Package ‘cplexAPI’

June 21, 2019

Type    Package
Title   R Interface to C API of IBM ILOG CPLEX
Version 1.3.6
Date    2019-06-21
Depends R (>= 2.6.0)
Imports methods
Description This is the R Interface to the C API of IBM ILOG CPLEX. It necessarily de-
           pends on IBM ILOG CPLEX (>= 12.1).
SystemRequirements IBM ILOG CPLEX (>= 12.1)
License GPL-3 | file LICENSE
LazyLoad yes
Collate  generics.R cplexConst.R cplexErrorClass.R cplexPtrClass.R
cplex.R cplexAPI.R cplex_checkAPI.R cplex_longparamAPI.R zzz.R
NeedsCompilation yes
Repository CRAN
Date/Publication 2019-06-21 21:50:04 UTC
Author Mayo Roettger [cre],
           Gabriel Gelius-Dietrich [aut],
           C. Jonathan Fritzemeier [ctb]
Maintainer Mayo Roettger <mayo.roettger@hhu.de>

R topics documented:
cplexAPI-package .................................................. 6
addColsCPLEX .................................................... 7
addFpDestCPLEX .................................................. 9
addIndConstrCPLEX .............................................. 10
addMIPstartsCPLEX .............................................. 11
addQConstrCPLEX ............................................... 12
addRowsCPLEX .................................................. 13
baroptCPLEX .................................................... 14
topics documented:

baseWriteCPLEX .................................................. 15
basicPresolveCPLEX ............................................. 16
boundSaCPLEX .................................................... 17
checkAddColsCPLEX ................................................ 18
checkAddRowsCPLEX ............................................... 19
checkChgCoefListCPLEX ......................................... 20
checkCopyColTypeCPLEX .......................................... 21
checkCopyLpCPLEX ................................................ 22
checkCopyLpwNamesCPLEX ....................................... 23
checkCopyQPsepCPLEX ........................................... 25
copyColsBndsCPLEX ............................................. 32
copyColTypeCPLEX ................................................ 33
copyMIPstartsCPLEX ............................................. 34
copyNameCPLEX ................................................... 35
copyObjCPLEX ..................................................... 36
copyProbNameCPLEX ............................................. 37
copyProbTypeCPLEX ............................................. 38
copyQPcoefCPLEX ............................................... 39
changeBaseCPLEX ................................................. 55
changeColNameCPLEX ............................................. 31
changeColsBndsCPLEX ........................................... 32
changeColTypeCPLEX ............................................. 33
changeMIPstartsCPLEX ......................................... 34
changeNameCPLEX ................................................. 35
changeObjCPLEX .................................................. 36
changeProbNameCPLEX .......................................... 37
changeProbTypeCPLEX .......................................... 38
changeQPcoefCPLEX ............................................. 39
changeRhsCPLEX .................................................. 40
changeRngValCPLEX ............................................. 41
changeRowNameCPLEX ........................................... 42
changeSenseCPLEX ............................................... 43
changeTerminateCPLEX ......................................... 44
cleanupCoefCPLEX .............................................. 44
copyBaseCPLEX ................................................... 51
copyColTypeCPLEX ................................................ 52
copyLpCPLEX ..................................................... 53
copyLpwNamesCPLEX ........................................... 54
copyObjNameCPLEX .............................................. 55
copyOrderCPLEX .................................................. 56
copyPartBaseCPLEX ............................................. 57
copyQPsepCPLEX .................................................. 58
copyQuadCPLEX ................................................... 59
copyStartCPLEX ................................................... 60
cplexConstants ..................................................... 61
cplexError-class ................................................. 83
R topics documented:

- cplexPtr-class ................................................. 84
- delColsCPLEX .................................................. 85
- delFpDestCPLEX ............................................... 86
- delIndConstrsCPLEX .......................................... 87
- delMIPstartsCPLEX ........................................... 88
- delNamesCPLEX ................................................ 89
- delProbCPLEX .................................................. 90
- delQConstrsCPLEX ............................................ 91
- delRowsCPLEX .................................................. 92
- delSetColsCPLEX ............................................. 93
- delSetRowsCPLEX ............................................. 94
- delTerminateCPLEX ........................................... 95
- disconnectChannelCPLEX ................................. 96
- dualoptCPLEX ................................................. 97
- dualWriteCPLEX ............................................. 98
- feasOptCPLEX ................................................. 99
- fileputCPLEX .................................................. 100
- flushChannelCPLEX ........................................... 101
- flushStdChannelsCPLEX ..................................... 102
- freePresolveCPLEX .......................................... 103
- getBaseCPLEX ................................................ 104
- getBestObjValCPLEX ......................................... 105
- getChannelsCPLEX ........................................... 106
- getChgParmCPLEX ............................................ 107
- getCoefCPLEX ................................................ 108
- getColIndexCPLEX ............................................ 109
- getColInfeasCPLEX .......................................... 110
- getColNameCPLEX ........................................... 111
- getColsCPLEX ................................................ 112
- getColTypeCPLEX ............................................ 113
- getConflictCPLEX ........................................... 114
- getConflictExtCPLEX ....................................... 115
- getCutoffCPLEX .............................................. 116
- getDb1ParmCPLEX ............................................ 117
- getDb1QualCPLEX ............................................ 118
- getDbsCntCPLEX ............................................. 119
- getDjCPLEX .................................................... 120
- getErrorStrCPLEX ............................................ 121
- getGradCPLEX ............................................... 122
- getIndConstrCPLEX ......................................... 123
- getInfoDb1ParmCPLEX ....................................... 124
- getInfoIntParmCPLEX ....................................... 125
- getInfoLongParmCPLEX ..................................... 126
- getInfoStrParmCPLEX ....................................... 127
- getIntParmCPLEX ............................................ 128
- getIntQualCPLEX ............................................ 129
- getIntCntCPLEX ............................................. 130
- getLogFileCPLEX ............................................ 131
### Topics Documented

- `getLogFileNameCPLEX` ................. 132
- `getLongParmCPLEX` ..................... 133
- `getLowBndsIdsCPLEX` ................... 134
- `getLowerBndsCPLEX` .................... 135
- `getMethodCPLEX` ....................... 136
- `getMIPrelGapCPLEX` .................... 137
- `getMIPstartIndexCPLEX` ............... 138
- `getMIPstartNameCPLEX` ............... 139
- `getMIPstartsCPLEX` ................. 140
- `getNumColsCPLEX` ..................... 141
- `getNumMIPstartsCPLEX` ............... 142
- `getNumNnzCPLEX` ...................... 143
- `getNumQConstrsCPLEX` ............... 144
- `getNumQPnzCPLEX` ..................... 145
- `getNumQuadCPLEX` ..................... 146
- `getNumRowsCPLEX` ..................... 147
- `getObjCPLEX` ......................... 148
- `getObjDirCPLEX` ....................... 149
- `getObjNameCPLEX` ...................... 150
- `getObjOffsetCPLEX` ................. 151
- `getObjValCPLEX` ....................... 152
- `getOrderCPLEX` ....................... 153
- `getParmHierNameCPLEX` ............... 154
- `getParmNameCPLEX` ..................... 155
- `getParmNumCPLEX` ..................... 156
- `getParmTypeCPLEX` ..................... 157
- `getParmValCPLEX` ...................... 158
- `getPhase1CntCPLEX` ................... 158
- `getPiCPLEX` ......................... 159
- `getPreStatCPLEX` ..................... 160
- `getProbNameCPLEX` ..................... 161
- `getProbTypeCPLEX` ..................... 162
- `getProbVarCPLEX` ..................... 163
- `getQConstrCPLEX` ..................... 164
- `getQPcoefCPLEX` ...................... 165
- `getQuadCPLEX` ....................... 166
- `getRedLpCPLEX` ....................... 167
- `getRhsCPLEX` ......................... 168
- `getRngValCPLEX` ...................... 169
- `getRowIndexCPLEX` ................... 170
- `getRowInfesCPLEX` .................... 171
- `getRowNameCPLEX` ..................... 172
- `getRowsCPLEX` ...................... 173
- `getSenseCPLEX` ....................... 174
- `getSiftItCntCPLEX` ................... 175
- `getSiftPase1CntCPLEX` ............... 176
- `getSlackCPLEX` ....................... 177
- `getStatCPLEX` ....................... 178
### Topics Documented

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>getStatStrCPLEX</td>
<td>179</td>
</tr>
<tr>
<td>getStrParmCPLEX</td>
<td>180</td>
</tr>
<tr>
<td>getSubMethodCPLEX</td>
<td>181</td>
</tr>
<tr>
<td>getSubStatCPLEX</td>
<td>182</td>
</tr>
<tr>
<td>getTimeCPLEX</td>
<td>183</td>
</tr>
<tr>
<td>getUppBndsIdsCPLEX</td>
<td>184</td>
</tr>
<tr>
<td>getUpperBndsCPLEX</td>
<td>185</td>
</tr>
<tr>
<td>getVersionCPLEX</td>
<td>186</td>
</tr>
<tr>
<td>hybbaroptCPLEX</td>
<td>186</td>
</tr>
<tr>
<td>hybmetoptCPLEX</td>
<td>187</td>
</tr>
<tr>
<td>initProbCPLEX</td>
<td>188</td>
</tr>
<tr>
<td>lpoptCPLEX</td>
<td>189</td>
</tr>
<tr>
<td>mipoptCPLEX</td>
<td>190</td>
</tr>
<tr>
<td>newColsCPLEX</td>
<td>191</td>
</tr>
<tr>
<td>newRowsCPLEX</td>
<td>192</td>
</tr>
<tr>
<td>objSaCPLEX</td>
<td>193</td>
</tr>
<tr>
<td>openEnvCPLEX</td>
<td>194</td>
</tr>
<tr>
<td>openFileCPLEX</td>
<td>195</td>
</tr>
<tr>
<td>openProbCPLEX</td>
<td>196</td>
</tr>
<tr>
<td>ordWriteCPLEX</td>
<td>197</td>
</tr>
<tr>
<td>preslvWriteCPLEX</td>
<td>198</td>
</tr>
<tr>
<td>presolveCPLEX</td>
<td>199</td>
</tr>
<tr>
<td>primoptCPLEX</td>
<td>200</td>
</tr>
<tr>
<td>printTerminateCPLEX</td>
<td>201</td>
</tr>
<tr>
<td>qpoptCPLEX</td>
<td>201</td>
</tr>
<tr>
<td>readCopyBaseCPLEX</td>
<td>202</td>
</tr>
<tr>
<td>readCopyMIPstartsCPLEX</td>
<td>203</td>
</tr>
<tr>
<td>readCopyOrderCPLEX</td>
<td>204</td>
</tr>
<tr>
<td>readCopyParmCPLEX</td>
<td>205</td>
</tr>
<tr>
<td>readCopyProbCPLEX</td>
<td>206</td>
</tr>
<tr>
<td>readCopySolCPLEX</td>
<td>207</td>
</tr>
<tr>
<td>refineConflictCPLEX</td>
<td>208</td>
</tr>
<tr>
<td>refineConflictExtCPLEX</td>
<td>209</td>
</tr>
<tr>
<td>refineMIPstartConflictCPLEX</td>
<td>210</td>
</tr>
<tr>
<td>refineMIPstartConflictExtCPLEX</td>
<td>211</td>
</tr>
<tr>
<td>return_codeCPLEX</td>
<td>212</td>
</tr>
<tr>
<td>rhsSaCPLEX</td>
<td>213</td>
</tr>
<tr>
<td>setDblParmCPLEX</td>
<td>214</td>
</tr>
<tr>
<td>setDefaultParmCPLEX</td>
<td>215</td>
</tr>
<tr>
<td>setIntParmCPLEX</td>
<td>216</td>
</tr>
<tr>
<td>setLogFileCPLEX</td>
<td>217</td>
</tr>
<tr>
<td>setLogFileNameCPLEX</td>
<td>218</td>
</tr>
<tr>
<td>setLongParmCPLEX</td>
<td>219</td>
</tr>
<tr>
<td>setObjDirCPLEX</td>
<td>220</td>
</tr>
<tr>
<td>setStrParmCPLEX</td>
<td>221</td>
</tr>
<tr>
<td>setTerminateCPLEX</td>
<td>222</td>
</tr>
<tr>
<td>siftoptCPLEX</td>
<td>223</td>
</tr>
<tr>
<td>solnInfoCPLEX</td>
<td>224</td>
</tr>
</tbody>
</table>
cplexAPI-package

Description
A low level interface to IBM ILOG CPLEX.

Details
The package cplexAPI provides access to the callable library of IBM ILOG CPLEX from within R.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

Examples
# load package
library(cplexAPI)

# Open a CPLEX environment
env <- openEnvCPLEX()

# Create a problem object
prob <- initProbCPLEX(env)

# Assign a name to the problem object
chgProbNameCPLEX(env, prob, "sample")

# Prepare data structures for the problem object
# Number of columns and rows
addColsCplexCplex

    nc <- 3
cr <- 3

    # Objective function
    obj <- c(5, 4, 3)

    # Right hand side
    rhs <- c(5, 11, 8)

    # Sense of the right hand side
    sense <- rep("L", 3)

    # Variable lower bounds
    lb <- rep(0, 3)

    # Variable upper bounds
    ub <- rep(CPX_INFBOUND, 3)

    # Column and row names
    cn <- c("x1", "x2", "x3")
    rn <- c("q1", "q2", "q3")

    # The constraint matrix is passed in column major order format
    # Be careful here: all indices start with 0! Begin indices of rows
    beg <- c(0, 3, 6)

    # Number of non-zero elements per row
    cnt <- rep(3, 3)

    # Column indices
    ind <- c(0, 1, 2, 0, 1, 2, 0, 1, 2)

    # Non-zero elements
    val <- c(2, 4, 3, 3, 1, 4, 1, 2, 2)

    # Load problem data
    copyLPwNamesCplex(env, prob, nc, nr, CPX_MAX, obj, rhs, sense,
                       beg, cnt, ind, val, lb, ub, NULL, cn, rn)

    # Solve the problem using the simplex algorithm
    lpoptCplex(env, prob)

    # Retrieve solution after optimization
    solutionCplex(env, prob)

    # Free memory, allacated to the problem object
    delProbCplex(env, prob)
closeEnvCplex(env)

addColsCplex  Adds Columns to a Specified CPLEX Problem Object
addColsCPLEX

Description
Low level interface function to the IBM ILOG CPLEX function CPXaddcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
addColsCPLEX(env, lp, ncols, nnz, objf, matbeg, matind, matval, 
lb = NULL, ub = NULL, cnames = NULL)

Arguments
env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols Number of columns.
nnz Number of nonzero constraint coefficients.
objf Objective function coefficients.
matbeg Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb Lower bounds of the new variables.
ub Upper bounds of the new variables.
cnames Names of the new variables.

Details
Interface to the C function addCols which calls the CPLEX function CPXaddcols.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**addFpDestCplex**

Add a File to the List of Message Destinations for a Channel

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxaddfpdest`. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where `cpxaddfpdest` has been removed.

**Usage**

```
addFpDestCplex(env, newch, cpfile)
```

**Arguments**

- **env**
  An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **newch**
  Pointer to an IBM ILOG CPLEX channel as returned by `addChannelCplex`.
- **cpfile**
  Pointer to an IBM ILOG CPLEX file as returned by `openFileCplex`.

**Details**

Interface to the C function `addFpDest` which calls the CPLEX function `CPXaddfpdest`.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

- Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- `delFpDestCplex`
addIndConstrcplex

Adds an Indicator Constraint to the Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddindconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```c
addIndConstrcplex(env, lp, indvar, complemented,
                   nzcnt, rhs, sense, linind, linval, indname = NULL)
```

Arguments

- `env`: An object of class "cplexPtr" as returned by openEnvcplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `indvar`: The binary variable that acts as the indicator for this constraint.
- `complemented`: A Boolean value that specifies whether the indicator variable is complemented.
- `nzcnt`: An integer that specifies the number of nonzero coefficients in the linear portion of the indicator constraint.
- `rhs`: The righthand side value for the linear portion of the indicator constraint.
- `sense`: The sense of the linear portion of the indicator constraint.
- `linind`: A vector that with `linval` defines the linear portion of the indicator constraint.
- `linval`: A vector that with `linind` defines the linear portion of the indicator constraint.
- `indname`: The name of the constraint to be added (optional).

Details

Interface to the C function addIndConstr which calls the CPLEX function CPXaddindconstr.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
addMIPstartsCPLEX  
Add Multiple MIP Starts to a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddmipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

addMIPstartsCPLEX(env, lp, mcnt, nzcnt, beg, varindices, values, effortlevel, mipstartname = NULL)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **mcnt**: Number of MIP starts to be added.
- **nzcnt**: Number of variable values to be added.
- **beg**: Array of length mcnt used with varindices and values. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **varindices**: Array of length nzcnt containing the numeric indices of the columns corresponding to the variables which are assigned starting values. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **values**: Array of length nzcnt containing the values to use for the MIP starts. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **effortlevel**: Array of length mcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **mipstartname**: Names of the MIP starts.

Details

Interface to the C function addMIPstarts which calls the CPLEX function CPXaddmipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
addQConstrCPLEX

Add Quadratic Constraint to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddqconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
addQConstrCPLEX(env, lp, lzn, qzn, rhs, sense,
                 lind = NULL, lval = NULL,
                 qrow, qcol, qval, qname = NULL)
```

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **lzn**: Number of nonzero constraint coefficients in the linear part of the constraint.
- **qzn**: Number of nonzero constraint coefficients in the quadratic part of the constraint.
- **rhs**: Right hand side term.
- **sense**: The sense of the constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **lind**: Linear part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **lval**: Linear part of the constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **qrow**: Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **qcol**: Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **qval**: Quadratic part of the quadratic constraint to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **qname**: Name of the constraint to be added.

Details

Interface to the C function addQConstr which calls the CPLEX function CPXaddqconstr.
Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

addRowsCplex

Add Constraints to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXaddrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

addRowsCplex(env, lp, ncols, nrows, nnz, matbeg, matind, matval,
              rhs = NULL, sense = NULL,
              cnames = NULL, rnames = NULL)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

ncols
Number of new columns in the constraints being added to the constraint matrix.

nrows
Number of rows.

nnz
Number of nonzero constraint coefficients.

matbeg
An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.

matind
An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.

matval
An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.

rhs
Righthand side term for each constraint to be added.

sense
Sense of each constraint to be added.

cnames
Names of the new columns.

rnames
Names of the new rows.
Details

Interface to the C function addCols which calls the CPLEX function CPXaddcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

ccheckAddRowsCplex, addColsCplex, copyLPcplex, chgRngValCplex

baroptCplex  Solve LP, QP or QCP Problem by Means of the Barrier Algorithm

Description

Low level interface function to the IBM ILOG CPLEX function CPXbaropt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

baroptCplex(env, lp)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function baropt which calls the CPLEX function CPXbaropt.

Value

Zero if successful, otherwise nonzero.
Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
solnInfoCPLEX, getStatCPLEX, solutionCPLEX

---

baseWriteCPLEX | Write the Most Current Basis Associated With a CPLEX Problem Object to a File

Description
Low level interface function to the IBM ILOG CPLEX function CPXmbasewrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
baseWriteCPLEX(env, lp, fname)

Arguments
- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **fname**: A filename.

Details
Interface to the C function baseWrite which calls the CPLEX function CPXmbasewrite.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

basicPresolveCPLEX

**Perform Bound Strengthening and Detect Redundant Rows**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXbasicpresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```
basicPresolveCPLEX(env, lp)
```

**Arguments**

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function basicPresolve which calls the CPLEX function CPXbasicpresolve.

**Value**

If successfull, a list will be returned:

- **redlb**: strengthened lower bounds
- **redub**: strengthened upper bounds
- **rstat**: status of the row

Otherwise an object of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
boundSaCPLEX

Access Upper and Lower Sensitivity Ranges for Lower and Upper Variable Bounds

Description

Low level interface function to the IBM ILOG CPLEX function CPXboundsa. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

boundSaCPLEX(env, lp, begin, end)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- begin: Beginning of the range of ranges to be returned.
- end: End of the range of ranges to be returned.

Details

Interface to the C function boundSa which calls the CPLEX function CPXboundsa.

Value

If successful, a list will be returned:

- lbLower: lower bound lower range values
- lbUpper: lower bound upper range values
- ubLower: upper bound lower range values
- ubUpper: upper bound upper range values

Otherwise an object of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
checkAddColsCplex  Validate Arguments of the Corresponding addColsCplex Routine

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckaddcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

checkAddColsCplex(env, lp, ncols, nnz, objf, matbeg, matind, matval, 
                  lb = NULL, ub = NULL, cnames = NULL)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp   An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
ncols Number of columns.
nnz  Number of nonzero constraint coefficients.
objf  Objective function coefficients.
matbeg Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval Array that specifies the nonzero elements of the columns being added. Consult the IBM ILOG CPLEX documentation for more detailed information.
lb    Lower bounds of the new variables.
ub    Upper bounds of the new variables.
cnames Names of the new variables.

Details

Interface to the C function checkAddCols which calls the CPLEX function CPXcheckaddcol.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hu.de>
checkAddRowsCplex

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

addColsCplex

---

**checkAddRowsCplex**

*Validate Arguments of the Corresponding addRowsCplex Routine*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXcheckaddrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
checkAddRowsCplex(env, lp, ncols, nrows, nnz, matbeg, matind, matval, 
    rhs = NULL, sense = NULL, 
    cnames = NULL, rnames = NULL)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `ncols` Number of new columns in the constraints being added to the constraint matrix.
- `nrows` Number of rows.
- `nnz` Number of nonzero constraint coefficients.
- `matbeg` An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matind` An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matval` An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `rhs` Righthand side term for each constraint to be added.
- `sense` Sense of each constraint to be added.
- `cnames` Names of the new columns.
- `rnames` Names of the new rows.

**Details**

Interface to the C function checkAddRows which calls the CPLEX function CPXcheckaddrows.
Value
Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
addRowsCplex

Description
Low level interface function to the IBM ILOG CPLEX function CPXcheckchgcoeflist. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
checkChgCoefListCplex(env, lp, nnz, ia, ja, ra)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
nnz Number of nonzero constraint coefficients.
ia Row indices of the nonzero elements.
ja Column indices of the nonzero elements.
ra Nonzero elements.

Details
Interface to the C function checkChgCoefList which calls the CPLEX function CPXcheckchgcoeflist.

Value
Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.
Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
chgCoefListCplex

Description
Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyctype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
checkCopyColTypeCplex(env, lp, xctype)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
xctype A vector containing the type of each column in the constraint matrix.

Details
Interface to the C function checkCopyColType which calls the CPLEX function CPXcheckcopyctype.

Value
Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

copyColTypeCplex

---

checkCopyLpCplex \_

Validate Arguments of the Corresponding copyLpCplex Routine

---

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopylp. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
checkCopyLpCplex(env, lp, nCols, nRows, lpdir, objf, rhs, sense,
                 matbeg, matcnt, matind, matval, lb, ub, rngval = NULL)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `nCols` Number of columns in the constraint matrix.
- `nRows` Number of rows in the constraint matrix.
- `lpdir` Single integer value that specifies whether the problem is a minimization or maximization problem.
- `objf` The objective function coefficients.
- `rhs` The righthand side values for each constraint in the constraint matrix.
- `sense` The sense of each constraint in the constraint matrix.
- `matbeg` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matcnt` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matind` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matval` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `lb` Containing the lower bound on each of the variables.
- `ub` Containing the lower bound on each of the variables.
- `rngval` Containing the range value of each ranged constraint.
Details

Interface to the C function checkCopyLp which calls the CPLEX function CPXcheckcopylp.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

copyLpCPLEX

checkCopyLpwNamesCLEX

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopylpwnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

checkCopyLpwNamesCLEX(env, lp, ncols, nrows, lpdir, objf, rhs, sense, matbeg, matcnt, matind, matval, lb, ub, rngval = NULL, cnames = NULL, rnames = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
<tr>
<td>ncols</td>
<td>Number of columns in the constraint matrix.</td>
</tr>
<tr>
<td>nrows</td>
<td>Number of rows in the constraint matrix.</td>
</tr>
<tr>
<td>lpdir</td>
<td>Single integer value that specifies whether the problem is a minimization or maximization problem.</td>
</tr>
</tbody>
</table>
**checkCopyLpwNamesCPLEX**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>objf</strong></td>
<td>The objective function coefficients.</td>
</tr>
<tr>
<td><strong>rhs</strong></td>
<td>The righthand side values for each constraint in the constraint matrix.</td>
</tr>
<tr>
<td><strong>sense</strong></td>
<td>The sense of each constraint in the constraint matrix.</td>
</tr>
<tr>
<td><strong>matbeg</strong></td>
<td>Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><strong>matcnt</strong></td>
<td>Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><strong>matind</strong></td>
<td>Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><strong>matval</strong></td>
<td>Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><strong>lb</strong></td>
<td>Containing the lower bound on each of the variables.</td>
</tr>
<tr>
<td><strong>ub</strong></td>
<td>Containing the lower bound on each of the variables.</td>
</tr>
<tr>
<td><strong>rngval</strong></td>
<td>Containing the range value of each ranged constraint.</td>
</tr>
<tr>
<td><strong>cnames</strong></td>
<td>Names of the matrix columns or, equivalently, the variable names.</td>
</tr>
<tr>
<td><strong>rnames</strong></td>
<td>Names of the matrix rows or, equivalently, the constraint names.</td>
</tr>
</tbody>
</table>

**Details**

Interface to the C function checkCopyLpwnames which calls the CPLEX function CPXcheckcopylpwnames.

**Value**

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

**See Also**

copyLpwnamesCPLEX
checkCopyQPsepCplex

Validate Arguments of the Corresponding copyQPsepCplex Routine

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyqpsep. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

checkCopyQPsepCplex(env, lp, qsepvec)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

qsepvec A vector containing the quadratic coefficients.

Details

Interface to the C function checkCopyQPsep which calls the CPLEX function CPXcheckcopyqpsep.

Value

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

copyQPsepCplex
checkCopyQuadCPLEX

Validate Arguments of the Corresponding checkCopyQuadCPLEX Routine

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXcheckcopyquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
checkCopyQuadCPLEX(env, lp, qmatbeg, qmatcnt, qmatind, qmatval)
```

**Arguments**

- `env`: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `qmatbeg`: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatcnt`: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatind`: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatval`: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Details**

Interface to the C function checkCopyQuad which calls the CPLEX function CPXcheckcopyquad.

**Value**

Nonzero if it detects an error in the data; it returns zero if it does not detect any data errors.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`copyQuadCPLEX`
checkValsCPELEX

Check an Array of Indices and a Corresponding Array of Values for Input Errors

Description

Low level interface function to the IBM ILOG CPLEX function CPXcheckvals. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

checkValsCPELEX(env, lp, nval, rind = NULL, cind = NULL, val = NULL)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPELEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPELEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

nval
Number of values to be examined.

rind
Row indices.

cind
Column indices.

val
The values itself.

Details

Interface to the C function checkVals which calls the CPLEX function CPXcheckvals.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgBndsCLEX  
*Change the Lower or Upper Bounds on a Set of Variables of a Problem*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXchgbnds. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```
chgBndsCLEX(env, lp, ncols, ind, lu, bd)
```

**Arguments**

- `env`: An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by initProbCLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `ncols`: Number of bounds to be changed.
- `ind`: Indices of bounds to be changed.
- `lu`: A character vector, specifying whether an entry in bd is a upper or a lower bound on variable ind[j].
- `bd`: Values of the lower or upper bounds of the variables present in ind.

**Details**

Interface to the C function chgBnds which calls the CPLEX function CPXchgbnds.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

getLowerBndsCLEX, getUpperBndsCLEX
chgCoefCPLEX

Change a Single Coefficient in the Constraint Matrix, Linear Objective Coefficients, Righthand Side, or Ranges of a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgCoefCPLEX(env, lp, i, j, val)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

i
An integer that specifies the numeric index of the row in which the coefficient is located. The linear objective row is referenced with i = -1.

j
An integer that specifies the numeric index of the column in which the coefficient is located. The RHS column is referenced with j = -1. The range value column is referenced with j = -2. If j = -2 is specified and row i is not a ranged row, an error status is returned.

val
The new value for the coefficient being changed.

Details

Interface to the C function chgCoef which calls the CPLEX function CPXchgcoef.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getNumRowsCPLEX, getNumColsCPLEX, chgObjCPLEX, chgRhsCPLEX, chgRngValCPLEX
chgCoefListCPLEX  Change a List of Matrix Coefficients of a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgcoeflist. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgCoefListCPLEX(env, lp, nnz, ia, ja, ra)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp   An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
nnz  Number of nonzero constraint coefficients.
ia   Row indices of the nonzero elements.
ja   Column indices of the nonzero elements.
ra   Nonzero elements.

Details

Interface to the C function chgcoeflist which calls the CPLEX function CPXchgcoeflist.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgColNameCPLEX

Change the Names of Variables in a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgcolname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgColNameCPLEX(env, lp, nnames, ind, names)

Arguments

env        An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp         An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
nnames     A vector that specifies the total number of variable names to be changed.
ind        A vector containing the numeric indices indices of the variables for which the names are to be changed.
names      A vector containing the strings of the new variable names for the columns specified in ind.

Details

Interface to the C function chgColName which calls the CPLEX function CPXchgcolname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgColsBndsCPLEX  

Change Lower and Upper Bounds on a Set of Variables of a Problem

Description
Set lower and upper bounds on a set of variables in one step. If \( lb[i] = ub[i] \) the type of the bound is set to "B", otherwise \( lb[i] \) is set to "L" and \( ub[i] \) is set to "U".

Usage
chgColsBndsCPLEX(env, lp, j, lb, ub)

Arguments
- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- j: An integer that specifies the numeric index of the column in which the coefficient is located.
- lb: A vector containing the lower bounds.
- ub: A vector containing the upper bounds.

Details
Interface to the C function chgColsBnds which calls the CPLEX function CPXchgbds.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
chgBndsCPLEX, tightenBndsCPLEX
chgColTypeCPLEX

Change Types of a Set of Variables of a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgctype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgColTypeCPLEX(env, lp, ncols, ind, xtype)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

ncols
Number of bounds to be changed.

ind
Indices of bounds to be changed.

xctype
A vector containing characters that represent the new types for the columns specified in indices.

Details

Interface to the C function chgColType which calls the CPLEX function CPXchgctype.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgMIPstartsCPELEX  Modify or Extend Multiple MIP Starts

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgmipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

    chgmipstartscplexHenvL lpL mcntL mipstartindicesL nzcntL
    begL varindicesL valuesL effortlevelI

Arguments

    env        An object of class "cplexPtr" as returned by openEnvCPELEX. This is basically
                a pointer to an IBM ILOG CPLEX environment.
    lp         An object of class "cplexPtr" as returned by initProbCPELEX. This is basically
                a pointer to an IBM ILOG CPLEX problem object.
    mcnt       Number of MIP starts to be changed.
    mipstartindices
                Array of length mcnt containing the numeric indices of the MIP starts to be
                changed.
    nzcnt      Number of entries to be changed.
    beg        Array of length mcnt used with varindices and values. Consult the IBM
                ILOG CPLEX documentation for more detailed information.
    varindices Array of length nzcnt containing the numeric indices of the columns corre-
                sponding to the variables which are assigned starting values. Consult the IBM
                ILOG CPLEX documentation for more detailed information.
    values     Array of length nzcnt containing the values to use for the MIP starts. Consult
                the IBM ILOG CPLEX documentation for more detailed information.
    effortlevel Array of length mcnt. Consult the IBM ILOG CPLEX documentation for more
                detailed information.

Details

Interface to the C function chgMIPstarts which calls the CPLEX function CPXchgmipstarts.

Value

    Zero if successful, otherwise nonzero.

Author(s)

    Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
    Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
chgNameCPLEX

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

chgNameCLEX \hspace{1em} Change the Name of a Constraint a Variable in a CPLEX Problem Object.

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgNameCLEX(env, lp, key, ij, name)

Arguments

- env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp   An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- key  A character to specify whether a row name or a column name should be changed.
- ij   An integer that specifies the numeric index of the column or row whose name is to be changed.
- name A pointer to a character string containing the new name.

Details

Interface to the C function chgName which calls the CPLEX function CPXchgname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**chgObjCPLEX**  
*Change Linear Objective Coefficients*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXchgobj. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```
chgobjcplex(env, lp, ncols, ind, val)
```

**Arguments**

- **env**
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- **lp**
  An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

- **ncols**
  Number of bounds to be changed.

- **ind**
  Indices of bounds to be changed.

- **val**
  A vector containing the new values of the objective coefficients of the variables specified in `ind`.

**Details**

Interface to the C function chgobj which calls the CPLEX function CPXchgobj.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

chgProbNameCplex

Change the Name of the Current Problem.

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgprobname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgProbNameCplex(env, lp, probname)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

probname
The new name of the problem.

Details

Interface to the C function chgProbName which calls the CPLEX function CPXchgprobname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgProbTypeCLEX  

*Change the Current Problem to a Related Problem*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXchgprobtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
chgProbTypeCLEX(env, lp, ptype)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `ptype` A single integer value specifying the problem type.

**Details**

Interface to the C function `chgProbType` which calls the CPLEX function CPXchgprobtype.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`getProbTypeCLEX`, `cplexConstants` section “Problem Types”.
chgQPcoefCPLEX

Change a Single Coefficient in the Quadratic Objective of a Quadratic Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgqpcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgQPcoefCLEX(env, lp, i, j, val)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **i**: The first variable number.
- **j**: The second variable number.
- **val**: The new coefficient value.

Details

Interface to the C function chgQPcoef which calls the CPLEX function CPXchgqpcoef.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

chgCoefCPLEX
chgRhsCPLEX  

Change Righthand Side Coefficients

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgrhs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgRhsCPLEX(env, lp, nrows, ind, val)

Arguments

env  
An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

nrows  
Number of bounds to be changed.

ind  
Indices of bounds to be changed.

val  
A vector containing the new values of the objective coefficients of the variables specified in ind.

Details

Interface to the C function chgRhs which calls the CPLEX function CPXchgrhs.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

Low level interface function to the IBM ILOG CPLEX function CPXchgrngval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgRngValCPLEX(env, lp, nrows, ind, val)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
nrows Number of bounds to be changed.
ind Indices of bounds to be changed.
val A vector containing the new values of the objective coefficients of the variables specified in ind.

Details

Interface to the C function chgRngVal which calls the CPLEX function CPXchgrngval.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgRowNameCPLEX     Change Names of Linear Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgRowName. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

chgRowNameCPLEX(env, lp, nnames, ind, names)

Arguments

env            An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically
               a pointer to an IBM ILOG CPLEX environment.
lp             An object of class "cplexPtr" as returned by initProbCPLEX. This is basically
               a pointer to an IBM ILOG CPLEX problem object.
nnames         A vector that specifies the total number of variable names to be changed.
ind            A vector containing the numeric indices indices of the variables for which the
               names are to be changed.
names          A vector containing the strings of the new variable names for the columns specified in ind.

Details

Interface to the C function chgRowName which calls the CPLEX function CPXchgRowName.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgSenseCplex 

**Change Sense of a Set of Linear Constraints**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXchgsense. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

`chgSenseCplex(env, lp, nrows, ind, sense)`

**Arguments**

- **env**: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **nrows**: Number of bounds to be changed.
- **ind**: Indices of bounds to be changed.
- **sense**: A vector containing characters that tell the new sense of the linear constraints specified in ind.

**Details**

Interface to the C function chgSense which calls the CPLEX function CPXchgsense.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
chgTerminateCplex  

Change Termination Signal

Description
The function chgTerminateCplex changes termination signal.

Usage
chgTerminateCplex(env, tval = 1)

Arguments
- env: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- tval: Single integer value.

Value
NULL

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
setTerminateCplex, delTerminateCplex, printTerminateCplex

cleanupCoeffCplex  

change Problem Coefficients to Zero That are Smaller in Magnitude Than the Tolerance Specified in the Argument eps

Description
Low level interface function to the IBM ILOG CPLEX function CPXcleanup. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
cleanupCoeffCplex(env, lp, eps)
cloneProbCplex

Arguments

- **env**: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **eps**: Single numeric value giving the tolerance.

Details

Interface to the C function `cleanupcoef` which calls the CPLEX function `CPXcleanup`.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

cloneProbCplex  

Copy a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function `CPXcloneprob`. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```r
cloneProbCplex(env, lp, ptrtype = "cplex_prob")
```

Arguments

- **env**: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **ptrtype**: A name for the pointer object.

Details

Interface to the C function `cloneProb` which calls the CPLEX function `CPXcloneprob`.
closeEnvCPLEX

Free all of the Data Structures Associated With CPLEX

Description
Low level interface function to the IBM ILOG CPLEX function CPXcloseCPLEX. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
closeEnvCPLEX(env)

Arguments
env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

Details
Interface to the C function closeEnv which calls the CPLEX function CPXcloseCPLEX.

Value
Zero if successful, otherwise an instance of class "cplexError".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
closeFileCPLEX

See Also

openEnvCPLEX

---

**closeFileCPLEX**  
*Close a File*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXfclose. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXfclose has been removed.

**Usage**

```r
closeFileCPLEX(cpfile)
```

**Arguments**

- `cpfile`  
  A pointer to a file as returned by `openFileCPLEX`.

**Details**

Interface to the C function cplexfclose which calls the CPLEX function CPXfclose.

**Value**

Zero if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

**See Also**

fileputCPLEX, openFileCPLEX
closeProbCPLEX  

Close CPLEX Environment And Remove CPLEX Problem Object

Description

The function closeProbCPLEX closes a CPLEX environment and removes a CPLEX problem object.

Usage

closeProbCPLEX(prob)

Arguments

prob  
A list containing a pointer to an IBM ILOG CPLEX environment and a Pointer to an IBM ILOG CPLEX problem object. Both elements are objects of class "cplexPtr" as returned by openProbCPLEX.

Details

Interface to the C functions delProb and closeEnv calling CPLEX functions CPXcloseCPLEX and CPXfreenprob.

Value

An integer vector containing the return values of CPXcloseCPLEX and CPXfreenprob.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

openProbCPLEX
cLpWriteCPLEX

Write an LP Format File Containing Identified Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXclpwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

\texttt{cLpWriteCPLEX(env, lp, fname)}

Arguments

\begin{itemize}
  \item \texttt{env} An object of class "\texttt{cplexPtr}" as returned by \texttt{openEnvCPLEX}. This is basically a pointer to an IBM ILOG CPLEX environment.
  \item \texttt{lp} An object of class "\texttt{cplexPtr}" as returned by \texttt{initProbCPLEX}. This is basically a pointer to an IBM ILOG CPLEX problem object.
  \item \texttt{fname} Single character value giving the filename to write to.
\end{itemize}

Details

Interface to the C function \texttt{cLpWriteCPLEX} which calls the CPLEX function CPXclpwrite.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at \url{https://www.ibm.com/support/knowledgcenter/SSSA5P}. 
completpCLEX  Manage Modification Steps Closely

Description

Low level interface function to the IBM ILOG CPLEX function CPXcompletpCLEX. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

completpCLEX(env, lp)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp   An object of class "cplexPtr" as returned by initProbCLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function completpCLEX which calls the CPLEX function CPXcompletpCLEX.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**copyBaseCplex**

Copies a Basis Into a CPLEX Problem Object.

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXcopybase. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
copyBaseCplex(env, lp, cstat, rstat)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `cstat` A vector containing the basis status of the columns in the constraint matrix.
- `rstat` A vector containing the basis status of the slack, or surplus, or artificial variable associated with each row in the constraint matrix.

**Details**

Interface to the C function `copyBase` which calls the CPLEX function CPXcopybase.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

copyColTypeCplex: Copy Variable Type Information Into a Given Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgprobnme. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```r
copyColTypeCplex(env, lp, xtype)
```

Arguments

- `env`: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `xctype`: A vector containing the type of each column in the constraint matrix.

Details

Interface to the C function `copyColType` which calls the CPLEX function CPXcopyctype.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopylp. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```c
copyLpCPLEX(env, lp, nCols, nRows, lpd, objf, rhs, sense,
matbeg, matcnt, matind, matval, lb, ub, rngval = NULL)
```

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **nCols**: Number of columns in the constraint matrix.
- **nRows**: Number of rows in the constraint matrix.
- **lpdir**: Single integer value that specifies whether the problem is a minimization or maximization problem.
- **objf**: The objective function coefficients.
- **rhs**: The righthand side values for each constraint in the constraint matrix.
- **sense**: The sense of each constraint in the constraint matrix.
- **matbeg**: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **matcnt**: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **matind**: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **matval**: Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **lb**: Containing the lower bound on each of the variables.
- **ub**: Containing the lower bound on each of the variables.
- **rngval**: Containing the range value of each ranged constraint.

Details

Interface to the C function copyLp which calls the CPLEX function CPXcopylp.
Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopylpwnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```r
copyLpwNamesCplex(env, lp, ncols, nrows, lpsense, objf, rhs, sense, matbeg, matcnt, matind, matval, lb, ub, 
rngval = NULL, cnames = NULL, rnames = NULL)
```

Arguments

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `ncols` Number of columns in the constraint matrix.
- `nrows` Number of rows in the constraint matrix.
- `lpsense` Single integer value that specifies whether the problem is a minimization or maximization problem.
- `objf` The objective function coefficients.
- `rhs` The righthand side values for each constraint in the constraint matrix.
- `sense` The sense of each constraint in the constraint matrix.
- `matbeg` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matcnt` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
copyObjNameCLEX

matind  Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval  Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
         Containing the lower bound on each of the variables.
ub      Containing the lower bound on each of the variables.
rngval  Containing the range value of each ranged constraint.
cnames  Names of the matrix columns or, equivalently, the variable names.
rnames  Names of the matrix rows or, equivalently, the constraint names.

Details
Interface to the C function copyLpWNames which calls the CPLEX function CPXcopylpwnames.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

copyObjNameCLEX  Copy a Name for the Objective Function Into a CPLEX Problem Object.

Description
Low level interface function to the IBM ILOG CPLEX function CPXcopyobjname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
copyObjNameCLEX(env, lp, oname)

Arguments
env     An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp      An object of class "cplexPtr" as returned by initProbCLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
oname   A pointer to a character string containing the objective name.
copyOrderCplex

Details

Interface to the C function copyObjName which calls the CPLEX function CPXcopyobjname.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

copyOrderCplex

Copy Priority Order to CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyorder. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

copyOrderCplex(env, lp, cnt, indices, priority = NULL, direction = NULL)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
cnt Number of entries.
indices Indices of the columns corresponding to the integer variables that are assigned priorities.
priority Priorities assigned to the integer variables.
direction Branching direction assigned to the integer variables.

Details

Interface to the C function copyOrder which calls the CPLEX function CPXcopyorder.
Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

copyPartBaseCPLEX  Copies a partial basis into an LP problem object.

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopypartialbase. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

copyPartBaseCPLEX(env, lp, ncind, cind, cstat, nrind, rind, rstat)

Arguments

env          An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp           An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
ncind        An integer that specifies the number of variable or column status values specified.
cind         A vector that contains the indices of the variables for which status values are being specified.
cstat        A vector where the ith entry contains the status for variable cind[i].
nrind        An integer that specifies the number of slack, surplus, or artificial status values specified.
rind         A vector rcnt that contains the indices of the slack, surplus, or artificial variables for which status values are being specified.
rstat        A vector of where the i-th entry contains the status for slack, surplus, or artificial rind[i].

Details

Interface to the C function copyPartBase which calls the CPLEX function CPXcopypartialbase.
Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

---

copyQPsepCPLEX  Copy the Quadratic Objective Matrix $Q$ for a Separable QP Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXcopyqpsep. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```r
copyQPsepCPLEX(env, lp, qsepvec)
```

Arguments

- **env**  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**  An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **qsepvec**  A vector containing the quadratic coefficients.

Details

Interface to the C function copyQPsep which calls the CPLEX function CPXcopyqpsep.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxcopyquad`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
copyQuadCLEX(env, lp, qmatbeg, qmatcnt, qmatind, qmatval)
```

**Arguments**

- `env` An object of class "`cplexPtr`" as returned by `openEnvCLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "`cplexPtr`" as returned by `initProbCLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `qmatbeg` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatcnt` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatind` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `qmatval` Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Details**

Interface to the C function `copyQuad` which calls the CPLEX function `CPXcopyquad`.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXcopystart. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
void copyStartCPLEX(Bcplexptr env, Bcplexptr lp, vec cstat, vec rstat, vec cprim, vec rprim, vec cdual, vec rdual);
```

**Arguments**

- **env** An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp** An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **cstat** A vector containing the basis status of the columns in the constraint matrix.
- **rstat** A vector containing the basis status of the slack, surplus, or artificial variable associated with each row in the constraint matrix.
- **cprim** A vector containing the initial primal values of the column variables.
- **rprim** A vector containing the initial primal values of the slack (row) variables.
- **cdual** A vector containing the initial values of the reduced costs for the column variables.
- **rdual** A vector containing the initial values of the dual variables for the rows.

**Details**

Interface to the C function copyStart which calls the CPLEX function CPXcopyStart.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

This is a list containing constants used by IBM ILOG CPLEX. Consult the IBM ILOG CPLEX manual for more information, in particular for the use of control parameters.

General Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_INFBOUND</td>
<td>1.0E+20</td>
</tr>
<tr>
<td>CPX_STR_PARAM_MAX</td>
<td>512</td>
</tr>
</tbody>
</table>

Types of parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_PARAMTYPE_NONE</td>
<td>0</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_INT</td>
<td>1</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_DOUBLE</td>
<td>2</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_STRING</td>
<td>3</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_LONG</td>
<td>4</td>
</tr>
</tbody>
</table>

Values returned for stat by solution

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_STAT_OPTIMAL</td>
<td>1</td>
</tr>
<tr>
<td>CPX_STAT_UNBOUNDED</td>
<td>2</td>
</tr>
<tr>
<td>CPX_STAT_INFEASIBLE</td>
<td>3</td>
</tr>
<tr>
<td>CPX_STAT_INFORUNBD</td>
<td>4</td>
</tr>
<tr>
<td>CPX_STAT_OPTIMAL_INFEAS</td>
<td>5</td>
</tr>
<tr>
<td>CPX_STAT_NUM_BEST</td>
<td>6</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_IT_LIM</td>
<td>10</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_TIME_LIM</td>
<td>11</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_OBJ_LIM</td>
<td>12</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_USER</td>
<td>13</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_SUM</td>
<td>14</td>
</tr>
<tr>
<td>CPX_STAT_OPTIMAL_RELAXED_SUM</td>
<td>15</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_INF</td>
<td>16</td>
</tr>
<tr>
<td>CPX_STAT_OPTIMAL_RELAXED_INF</td>
<td>17</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_QUAD</td>
<td>18</td>
</tr>
</tbody>
</table>
Solution type return values from CPXsolninfo

- CPX_NO_SOLN = 0
- CPX_BASIC_SOLN = 1
- CPX_NONBASIC_SOLN = 2
- CPX_PRIMAL_SOLN = 3

Values of presolve stats for columns and rows

- CPX_PRECOL_LOW = -1: fixed to original lb
- CPX_PRECOL_UP = -2: fixed to original ub
- CPX_PRECOL_FIX = -3: fixed to some other value
- CPX_PRECOL_AGG = -4: aggregated $y = a \times x + b$
- CPX_PRECOL_OTHER = -5: cannot be expressed by a linear combination of active variables in the presolved model
- CPX_PREROW_RED = -1: redundant row removed in presolved model
- CPX_PREROW_AGG = -2: used to aggregate a variable
- CPX_PREROW_OTHER = -3: other, for example merge two inequalities into a single equation

Generic constants

- CPX_ON = 1
- CPX_OFF = 0
- CPX_MAX = -1
- CPX_MIN = 1

Primal simplex pricing algorithm

- CPX_PPRIIND_PARTIAL = -1
- CPX_PPRIIND_AUTO = 0
- CPX_PPRIIND_DEVEX = 1
cplexConstants

CPX_PPRIIND_STEEP  2
CPX_PPRIIND_STEEPQSTART  3
CPX_PPRIIND_FULL  4

Dual simplex pricing algorithm

CPX_DPRIIND_AUTO  0
CPX_DPRIIND_FULL  1
CPX_DPRIIND_STEEP  2
CPX_DPRIIND_FULL_STEEP  3
CPX_DPRIIND_STEEPQSTART  4
CPX_DPRIIND_DEVEX  5

PARALLELMODE values

CPX_PARALLEL_DETERMINISTIC  1
CPX_PARALLEL_AUTO  0
CPX_PARALLEL_OPPORTUNISTIC -1

Values for CPX_PARAM_WRITELEVEL

CPX_WRITELEVEL_AUTO  0
CPX_WRITELEVEL_ALLVARS  1
CPX_WRITELEVEL_DISCRETEVARS  2
CPX_WRITELEVEL_NONZEROVARS  3
CPX_WRITELEVEL_NONZERODISCRETEVARS  4

Values for CPX_PARAM_SOLUTIONTARGET

CPX_SOLUTIONTARGET_AUTO  0
CPX_SOLUTIONTARGET_OPTIMALCONVEX  1
CPX_SOLUTIONTARGET_FIRSTORDER  2
CPX_SOLUTIONTARGET_OPTIMALGLOBAL  3
LP/QP solution algorithms

Used as possible values for CPX_PARAM_LPMETHOD, CPX_PARAM_QPMETHOD, CPX_PARAM_BARCROSSALG, CPXgetmethod, ...

<table>
<thead>
<tr>
<th>cplexAlgConstant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_ALG_NONE</td>
<td>-1</td>
</tr>
<tr>
<td>CPX_ALG_AUTOMATIC</td>
<td>0</td>
</tr>
<tr>
<td>CPX_ALG_PRIMAL</td>
<td>1</td>
</tr>
<tr>
<td>CPX_ALG_DUAL</td>
<td>2</td>
</tr>
<tr>
<td>CPX_ALG_NET</td>
<td>3</td>
</tr>
<tr>
<td>CPX_ALG_BARRIER</td>
<td>4</td>
</tr>
<tr>
<td>CPX_ALG_SIFTING</td>
<td>5</td>
</tr>
<tr>
<td>CPX_ALG_CONCURRENT</td>
<td>6</td>
</tr>
<tr>
<td>CPX_ALG_BAROPT</td>
<td>7</td>
</tr>
<tr>
<td>CPX_ALG_PIVOTIN</td>
<td>8</td>
</tr>
<tr>
<td>CPX_ALG_PIVOTOUT</td>
<td>9</td>
</tr>
<tr>
<td>CPX_ALG_PIVOT</td>
<td>10</td>
</tr>
<tr>
<td>CPX_ALG_FEASOPT</td>
<td>11</td>
</tr>
<tr>
<td>CPX_ALG_MIP</td>
<td>12</td>
</tr>
<tr>
<td>CPX_ALG_ROBUST</td>
<td>13</td>
</tr>
</tbody>
</table>

Basis status values

<table>
<thead>
<tr>
<th>cplexBasisConstant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_AT_LOWER</td>
<td>0</td>
</tr>
<tr>
<td>CPX_BASIC</td>
<td>1</td>
</tr>
<tr>
<td>CPX_AT_UPPER</td>
<td>2</td>
</tr>
<tr>
<td>CPX_FREE_SUPER</td>
<td>3</td>
</tr>
</tbody>
</table>

Variable types for ctype array

<table>
<thead>
<tr>
<th>cplexTypeConstant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_CONTINUOUS</td>
<td>&quot;C&quot;</td>
</tr>
<tr>
<td>CPX_BINARY</td>
<td>&quot;B&quot;</td>
</tr>
<tr>
<td>CPX_INTEGER</td>
<td>&quot;I&quot;</td>
</tr>
<tr>
<td>CPX_SEMICONT</td>
<td>&quot;S&quot;</td>
</tr>
<tr>
<td>CPX_SEMIINT</td>
<td>&quot;N&quot;</td>
</tr>
</tbody>
</table>

PREREDUCE settings
Conflict statuses

CPX_STAT_CONFLICT_FEASIBLE 30
CPX_STAT_CONFLICT_MINIMAL 31
CPX_STAT_CONFLICT_ABORT_CONTRADICTION 32
CPX_STAT_CONFLICT_ABORT_TIME_LIM 33
CPX_STAT_CONFLICT_ABORT_IT_LIM 34
CPX_STAT_CONFLICT_ABORT_NODE_LIM 35
CPX_STAT_CONFLICT_ABORT_OBJ_LIM 36
CPX_STAT_CONFLICT_ABORT_MEM_LIM 37
CPX_STAT_CONFLICT_ABORT_USER 38
CPX_STAT_CONFLICT_ABORT_DETTIME_LIM 39

Conflict status values

CPX_CONFLICT_EXCLUDED -1
CPX_CONFLICT_POSSIBLE_MEMBER 0
CPX_CONFLICT_POSSIBLE_LB 1
CPX_CONFLICT_POSSIBLE_UB 2
CPX_CONFLICT_MEMBER 3
CPX_CONFLICT_LB 4
CPX_CONFLICT_UB 5

Problem Types

Types 4, 9, and 12 are internal, the others are for users.

CPXPROB_LP 0
CPXPROB_MILP 1
CPXPROB_FIXEDMILP 3
CPXPROB_NODELP 4
CPXPROB_QP 5
CPXPROB_MIQP 7
CPLEX Parameter numbers

CPXPARAM_ADVIND 1001
CPXPARAM_AGGFILL 1002
CPXPARAM_AGGIND 1003
CPXPARAM_BASINTERVAL 1004
CPXPARAM_CFILEMUL 1005
CPXPARAM_CLOCKTYPE 1006
CPXPARAM_CRAINDD 1007
CPXPARAM_DEPIND 1008
CPXPARAM_DPRIIND 1009
CPXPARAM_PRICELIM 1010
CPXPARAM_EPMRK 1013
CPXPARAM_EPOPT 1014
CPXPARAM_EPPER 1015
CPXPARAM_EPRHS 1016
CPXPARAM_FASTMIP 1017
CPXPARAM_SIMDISPLAY 1019
CPXPARAM_ITLIM 1020
CPXPARAM_ROWREADLIM 1021
CPXPARAM_NETFIND 1022
CPXPARAM_COLREADLIM 1023
CPXPARAM_NZREADLIM 1024
CPXPARAM_OBJLIM 1025
CPXPARAM_OBJULIM 1026
CPXPARAM_PERIND 1027
CPXPARAM_PERLIM 1028
CPXPARAM_PPRIIND 1029
CPXPARAM_PREIND 1030
CPXPARAM_REINV 1031
CPXPARAM_REVERSEIND 1032
CPXPARAM_RFFILEMUL 1033
CPXPARAM_SCAIND 1034
CPXPARAM_SCRIND 1035
CPXPARAM_SINGLIM 1037
CPXPARAM_SINGTOL 1038
CPXPARAM_TILIM 1039
CPXPARAM_XXXIND 1041
CPXPARAM_PREDUAL 1044
CPXPARAM_EPOPT_H 1049
CPXPARAM_EPRHS_H 1050
cplexConstants

CPX_PARAM_PREPASS 1052
CPX_PARAM_DATACHECK 1056
CPX_PARAM_REDUCE 1057
CPX_PARAM_PRELINEAR 1058
CPX_PARAM_LPMETHOD 1062
CPX_PARAM_QPMETHOD 1063
CPX_PARAM_WORKDIR 1064
CPX_PARAM_WORKMEM 1065
CPX_PARAM_THREADS 1067
CPX_PARAM_CONFLICTDISPLAY 1074
CPX_PARAM_SIFTDISPLAY 1076
CPX_PARAM_SIFTALG 1077
CPX_PARAM_SIFTITLIM 1078
CPX_PARAM_MPSLONGNUM 1081
CPX_PARAM_MEMORYEMPHASIS 1082
CPX_PARAM_NUMERICALEMPHASIS 1083
CPX_PARAM_FEASOPTMODE 1084
CPX_PARAM_PARALLELMODE 1109
CPX_PARAM_TUNINGMEASURE 1110
CPX_PARAM_TUNINGREPEAT 1111
CPX_PARAM_TUNINGITLIM 1112
CPX_PARAM_TUNINGDISPLAY 1113
CPX_PARAM_WRITELEVEL 1114
CPX_PARAM_RANDOMSEED 1124
CPX_PARAM_DETITLIM 1127
CPX_PARAM_FILEENCODING 1129
CPX_PARAM_APIENCODING 1130
CPX_PARAM_SOLUTIONTARGET 1131
CPX_PARAM_CLONELOG 1132
CPX_PARAM_TUNINGDETITLIM 1139
CPX_PARAM_ALL_MIN 1000
CPX_PARAM_ALL_MAX 6000

Values for CPX_PARAM_TUNINGMEASURE

CPX_TUNE_AVERAGE 1
CPX_TUNE_MINMAX 2

Values for incomplete tuning

CPX_TUNE_ABORT 1
Quality query identifiers

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_MAX_PRIMAL_INFEAS</td>
<td>1</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_PRIMAL_INFEAS</td>
<td>2</td>
</tr>
<tr>
<td>CPX_SUM_PRIMAL_INFEAS</td>
<td>3</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_PRIMAL_INFEAS</td>
<td>4</td>
</tr>
<tr>
<td>CPX_MAX_DUAL_INFEAS</td>
<td>5</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_DUAL_INFEAS</td>
<td>6</td>
</tr>
<tr>
<td>CPX_SUM_DUAL_INFEAS</td>
<td>7</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_DUAL_INFEAS</td>
<td>8</td>
</tr>
<tr>
<td>CPX_MAX_INT_INFEAS</td>
<td>9</td>
</tr>
<tr>
<td>CPX_SUM_INT_INFEAS</td>
<td>10</td>
</tr>
<tr>
<td>CPX_MAX_PRIMAL_RESIDUAL</td>
<td>11</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_PRIMAL_RESIDUAL</td>
<td>12</td>
</tr>
<tr>
<td>CPX_SUM_PRIMAL_RESIDUAL</td>
<td>13</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_PRIMAL_RESIDUAL</td>
<td>14</td>
</tr>
<tr>
<td>CPX_MAX_DUAL_RESIDUAL</td>
<td>15</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_DUAL_RESIDUAL</td>
<td>16</td>
</tr>
<tr>
<td>CPX_SUM_DUAL_RESIDUAL</td>
<td>17</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_DUAL_RESIDUAL</td>
<td>18</td>
</tr>
<tr>
<td>CPX_MAX_COMP_SLACK</td>
<td>19</td>
</tr>
<tr>
<td>CPX_MAX_X</td>
<td>23</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_X</td>
<td>24</td>
</tr>
<tr>
<td>CPX_MAX_PI</td>
<td>25</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_PI</td>
<td>26</td>
</tr>
<tr>
<td>CPX_MAX_SLACK</td>
<td>27</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_SLACK</td>
<td>28</td>
</tr>
<tr>
<td>CPX_MAX_RED_COST</td>
<td>29</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_RED_COST</td>
<td>30</td>
</tr>
<tr>
<td>CPX_SUM_X</td>
<td>31</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_X</td>
<td>32</td>
</tr>
<tr>
<td>CPX_SUM_PI</td>
<td>33</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_PI</td>
<td>34</td>
</tr>
<tr>
<td>CPX_SUM_SLACK</td>
<td>35</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_SLACK</td>
<td>36</td>
</tr>
<tr>
<td>CPX_SUM_RED_COST</td>
<td>37</td>
</tr>
<tr>
<td>CPX_SUM_SCALED_RED_COST</td>
<td>38</td>
</tr>
<tr>
<td>CPX_KAPPA</td>
<td>39</td>
</tr>
<tr>
<td>CPX_OBJ_GAP</td>
<td>40</td>
</tr>
<tr>
<td>CPX_DUAL_OBJ</td>
<td>41</td>
</tr>
<tr>
<td>CPX_PRIMAL_OBJ</td>
<td>42</td>
</tr>
</tbody>
</table>
cplexConstants

CPX_MAX_QCPRIMAL_RESIDUAL 43
CPX_SUM_QCPRIMAL_RESIDUAL 44
CPX_MAX_QCSLACK_INFEAS 45
CPX_SUM_QCSLACK_INFEAS 46
CPX_MAX_QCSLACK 47
CPX_SUM_QCSLACK 48
CPX_MAX_INDSLACK_INFEAS 49
CPX_SUM_INDSLACK_INFEAS 50
CPX_EXACT_KAPPA 51
CPX_KAPPA_STABLE 52
CPX_KAPPA_SUSPICIOUS 53
CPX_KAPPA_UNSTABLE 54
CPX_KAPPA_HLLPOSED 55
CPX_KAPPA_MAX 56
CPX_KAPPA_ATTENTION 57

Solution quality symbols new in CPLEX 12.9.0

CPX_MAX_PWLSLACK_INFEAS 58
CPX_SUM_PWLSLACK_INFEAS 59

feasopt options

CPX_FEASOPT_MIN_SUM 0
CPX_FEASOPT_OPT_SUM 1
CPX_FEASOPT_MIN_INF 2
CPX_FEASOPT_OPT_INF 3
CPX_FEASOPT_MIN_QUAD 4
CPX_FEASOPT_OPT_QUAD 5

File: barconst.h

CPX_STAT_OPTIMAL_FACE_UNBOUNDED 20
CPX_STAT_ABORT_PRIM_OBJ_LIM 21
CPX_STAT_ABORT_DUAL_OBJ_LIM 22
CPX_STAT_FIRSTORDER 24
Barrier parameters

CPX_PARAM_BARDSTART 3001
CPX_PARAM_BAREPCOMP 3002
CPX_PARAM_BARGROWTH 3003
CPX_PARAM_BAROBJRNG 3004
CPX_PARAM_BARPSTART 3005
CPX_PARAM_BARALG 3007
CPX_PARAM_BARCOLNZ 3009
CPX_PARAM_BARDISPLAY 3010
CPX_PARAM_BARITLIM 3012
CPX_PARAM_BARMAXCOR 3013
CPX_PARAM_BARORDER 3014
CPX_PARAM_BARSTARTALG 3017
CPX_PARAM_BARCROSSALG 3018
CPX_PARAM_BARQCPEEDCOMP 3020

Optimizing Problems

CPX_BARORDER_AUTO 0
CPX_BARORDER_AMD 1
CPX_BARORDER_AMF 2
CPX_BARORDER_ND 3

MIP emphasis settings

CPX_MIEMPHASIS_BALANCED 0
CPX_MIEMPHASIS_FEASIBILITY 1
CPX_MIEMPHASIS_OPTIMALITY 2
CPX_MIEMPHASIS_BESTBOUND 3
CPX_MIEMPHASIS_HIDDENFEAS 4

Values for sostype and branch type

CPX_TYPE_VAR "0"
cplexConstants

CPX_TYPE_SOS1 "1"
CPX_TYPE_SOS2 "2"
CPX_TYPE_USER "X"
CPX_TYPE_ANY "A"

Variable selection values

CPX_VARSEL_MININFEAS -1
CPX_VARSEL_DEFAULT  0
CPX_VARSEL_MAXINFEAS  1
CPX_VARSEL_PSEUDO    2
CPX_VARSEL_STRONG    3
CPX_VARSEL_PSEUDOREDUCTED 4

Node selection values

CPX_NODESEL_DFS     0
CPX_NODESEL_BESTBOUND  1
CPX_NODESEL_BESTEST  2
CPX_NODESEL_BESTEST_ALT 3

Values for generated priority order

CPX_MIPORDER_COST     1
CPX_MIPORDER_BOUNDS   2
CPX_MIPORDER_SCALED_COST 3

Values for direction array

CPX_BRANCH_GLOBAL  0
CPX_BRANCH_DOWN    -1
CPX_BRANCH_UP      1
Values for **CPX_PARAM_BRDIR**

- CPX_BRDIR_DOWN  -1
- CPX_BRDIR_AUTO   0
- CPX_BRDIR_UP     1

Values for **CPX_PARAM_MIPSEARCH**

- CPX_MIPSEARCH_AUTO     0
- CPX_MIPSEARCH_TRADITIONAL 1
- CPX_MIPSEARCH_DYNAMIC  2

Values for **CPX_PARAM_MIPKAPPASTATS**

- CPX_MIPKAPPA_OFF       -1
- CPX_MIPKAPPA_AUTO      0
- CPX_MIPKAPPA_SAMPLE    1
- CPX_MIPKAPPA_FULL      2

Effort levels for MIP starts

- CPX_MIPSTART_AUTO    0
- CPX_MIPSTART_CHECKFEAS 1
- CPX_MIPSTART_SOLVEFIXED 2
- CPX_MIPSTART_SOLVEMIP  3
- CPX_MIPSTART_REPAIR   4

MIP Problem status codes

- CPXMIP_OPTIMAL       101
- CPXMIP_OPTIMAL_TOL   102
Valid purgeable values for adding usercuts and lazyconstraints

    CPX_USECUT_FORCE  0
    CPX_USECUT_PURGE   1
    CPX_USECUT_FILTER  2

For CPXgetnodeintfeas

    CPX_INTEGER_FEASIBLE  0
    CPX_INTEGER_INFEASIBLE  1
MIP Parameter numbers

CPX_PARAM_BRDIR  2001
CPX_PARAM_BTTL0  2002
CPX_PARAM_CLIQUEs  2003
CPX_PARAM_COEREIND  2004
CPX_PARAM_COVERS  2005
CPX_PARAM_CUTLO  2006
CPX_PARAM_CUTUP  2007
CPX_PARAM_EPGAAP  2008
CPX_PARAM_EPGAP  2009
CPX_PARAM_EPINT  2010
CPX_PARAM_MIPDISPLAY  2012
CPX_PARAM_MIPINTERVAL  2013
CPX_PARAM_INTSOLLIM  2015
CPX_PARAM_NODEFILEIND  2016
CPX_PARAM_NODELIM  2017
CPX_PARAM_NODESEL  2018
CPX_PARAM_OBJDIFF  2019
CPX_PARAM_MIPORDIND  2020
CPX_PARAM_RELOBJDIF  2022
CPX_PARAM_STARTALG  2025
CPX_PARAM_SUBALG  2026
CPX_PARAM_TRELIM  2027
CPX_PARAM_VARSEL  2028
CPX_PARAM_BNDSTREIND  2029
CPX_PARAM_HEURFREQ  2031
CPX_PARAM_MIPORDTYPE  2032
CPX_PARAM_CUTSFACOR  2033
CPX_PARAM_RELAXPREIND  2034
CPX_PARAM_PRESLVND  2037
CPX_PARAM_BBINTERVAL  2039
CPX_PARAM_FLOWCOVERS  2040
CPX_PARAM_IMPLBD  2041
CPX_PARAM_PROBE  2042
CPX_PARAM_GUBCOVERS  2044
CPX_PARAM_STRONGCANLIM  2045
CPX_PARAM_STRONGITLIM  2046
CPX_PARAM_FRACCAND  2048
CPX_PARAM_FRACCUTS  2049
CPX_PARAM_FRACPASS  2050
CPX_PARAM_FLOWPATHS  2051
CPX_PARAM_MIRCUTS  2052
Values for CPX_PARAM_SOLNPOOLREPLACE
### cplexConstants

**CPX_SOLNPOOL_FIFO** 0
**CPX_SOLNPOOL_OBJ** 1
**CPX_SOLNPOOL_DIV** 2
**CPX_SOLNPOOL_FILTER_DIVERSITY** 1
**CPX_SOLNPOOL_FILTER_RANGE** 2

**File:** gcconst.h

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_CON_LOWER_BOUND</td>
<td>1</td>
</tr>
<tr>
<td>CPX_CON_UPPER_BOUND</td>
<td>2</td>
</tr>
<tr>
<td>CPX_CON_LINEAR</td>
<td>3</td>
</tr>
<tr>
<td>CPX_CON_QUADRATIC</td>
<td>4</td>
</tr>
<tr>
<td>CPX_CON_SOS</td>
<td>5</td>
</tr>
<tr>
<td>CPX_CON_INDICATOR</td>
<td>6</td>
</tr>
</tbody>
</table>

**internal types**

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_CON_MINEXPR</td>
<td>7</td>
</tr>
<tr>
<td>CPX_CON_MAXEXPR</td>
<td>8</td>
</tr>
<tr>
<td>CPX_CON_PWL</td>
<td>9</td>
</tr>
<tr>
<td>CPX_CON_ABS</td>
<td>9</td>
</tr>
<tr>
<td>CPX_CON_DISJCST</td>
<td>10</td>
</tr>
<tr>
<td>CPX_CON_INDDISJCST</td>
<td>11</td>
</tr>
<tr>
<td>CPX_CON_SETVAR</td>
<td>12</td>
</tr>
<tr>
<td>CPX_CON_SETVARMEMBER</td>
<td>13</td>
</tr>
<tr>
<td>CPX_CON_SETVARCARD</td>
<td>14</td>
</tr>
<tr>
<td>CPX_CON_SETVARSUM</td>
<td>15</td>
</tr>
<tr>
<td>CPX_CON_SETVARMIN</td>
<td>16</td>
</tr>
<tr>
<td>CPX_CON_SETVARMAX</td>
<td>17</td>
</tr>
<tr>
<td>CPX_CON_SETVARSUBSET</td>
<td>18</td>
</tr>
<tr>
<td>CPX_CON_SETVARDOMAIN</td>
<td>19</td>
</tr>
<tr>
<td>CPX_CON_SETVARUNION</td>
<td>20</td>
</tr>
<tr>
<td>CPX_CON_SETVARINTERSECTION</td>
<td>21</td>
</tr>
<tr>
<td>CPX_CON_SETVARNULLINTERSECT</td>
<td>22</td>
</tr>
<tr>
<td>CPX_CON_SETVARIINTERSECT</td>
<td>23</td>
</tr>
<tr>
<td>CPX_CON_SETVAREQ</td>
<td>24</td>
</tr>
<tr>
<td>CPX_CON_SETVARNEQ</td>
<td>25</td>
</tr>
<tr>
<td>CPX_CON_SETVARNEQ_CST</td>
<td>26</td>
</tr>
<tr>
<td>CPX_CON_LAST_CONTYPE</td>
<td>27</td>
</tr>
</tbody>
</table>
Network parameters

CPX_PARAM_NETITLIM  5001
CPX_PARAM_NETEPOPT  5002
CPX_PARAM_NETWORKS  5003
CPX_PARAM_NETPRININD  5004
CPX_PARAM_NETDISPLAY  5005

NETOPT display values

CPXNET_NO_DISPLAY_OBJECTIVE  0
CPXNET_TRUE_OBJECTIVE  1
CPXNET_PENALIZED_OBJECTIVE  2

NETOPT pricing parameters

CPXNET_PRICE_AUTO  0
CPXNET_PRICE_PARTIAL  1
CPXNET_PRICE_MULT_PART  2
CPXNET_PRICE_SORT_MULT_PART  3

Copying data

CPX_PARAM_QPNZREADLIM  4001

Specify how to calculate duals for QCPs

CPX_PARAM_CALCQCPDUALS  4003
presolve

CPX_PARAM_QPMAKEPSDIND  4010

Error codes
Callable library miscellaneous routines

CPXERR_NEGATIVE_SURPLUS  1207
CPXERR_NO_SENSIT  1260

Error codes new in CPLEX 12.8.0
Callable library miscellaneous routines

CPXERR_CALLBACK_INCONSISTENT  1060
CPXERR_CAND_NOTRAY  3026
CPXERR_CAND_NOTPOINT  3025

Error codes new in CPLEX 12.9.0
Callable library miscellaneous routines

CPXERR_BAD_MULTIOBJ_ATTR  1488
CPXERR_MULTIOBJ_SUBPROBSOLVE  1300
CPXERR_NO_OBJ_NAME  1486
CPXERR_NOT_FOR_MULTIOBJ  1070

new parameter names introduced in IBM ILOG CPLEX version 12.6
Callable library miscellaneous routines

CPXPARAM_Advance  1001
CPXPARAM_BARRIER_Algorithm  3007
CPXPARAM_BARRIER_ColNonzeros  3009
CPXPARAM_BARRIER_ConvergeTol  3002
CPXPARAM_BARRIER_Crossover  3018
CPXPARAM_BARRIER_Display  3010
CPXPARAM_BARRIER_LIMITS_Corrections  3013
CPXPARAM_BARRIER_LIMITS_Growth  3003
CPXPARAM_BARRIER_LIMITS_Iteration  3012
CPXPARAM_BARRIER_LIMITS_ObjRange  3004
CPXPARAM_BARRIER_Ordering  3014
<table>
<thead>
<tr>
<th>Constant Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPXPARAM_Barrier_QCPConvergeTol</td>
<td>3020</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_StartAlg</td>
<td>3017</td>
</tr>
<tr>
<td>CPXPARAM_ClockType</td>
<td>1006</td>
</tr>
<tr>
<td>CPXPARAM_Conflict_Display</td>
<td>1074</td>
</tr>
<tr>
<td>CPXPARAM_DetTimeLimit</td>
<td>1127</td>
</tr>
<tr>
<td>CPXPARAM_DistMIP_Rampup_DetTimeLimit</td>
<td>2164</td>
</tr>
<tr>
<td>CPXPARAM_DistMIP_Rampup_Duration</td>
<td>2163</td>
</tr>
<tr>
<td>CPXPARAM_DistMIP_Rampup_TimeLimit</td>
<td>2165</td>
</tr>
<tr>
<td>CPXPARAM_Emphasis_Memory</td>
<td>1082</td>
</tr>
<tr>
<td>CPXPARAM_Emphasis_MIP</td>
<td>2058</td>
</tr>
<tr>
<td>CPXPARAM_Emphasis_Numerical</td>
<td>1083</td>
</tr>
<tr>
<td>CPXPARAM_Feasopt_Mode</td>
<td>1084</td>
</tr>
<tr>
<td>CPXPARAM_Feasopt_Tolerance</td>
<td>2073</td>
</tr>
<tr>
<td>CPXPARAM_LPMethod</td>
<td>1062</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_Cliques</td>
<td>2003</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_Covers</td>
<td>2005</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_Disjunctive</td>
<td>2053</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_FlowCovers</td>
<td>2040</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_Gomory</td>
<td>2049</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_GUBCovers</td>
<td>2044</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_Implied</td>
<td>2041</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_LiftProj</td>
<td>2152</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_MCFCut</td>
<td>2134</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_MIRCut</td>
<td>2052</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_PathCut</td>
<td>2051</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Cuts_ZeroHalfCut</td>
<td>2111</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Display</td>
<td>2012</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Interval</td>
<td>2013</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_AggForCut</td>
<td>2054</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_AuxRootThreads</td>
<td>2139</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_CutPasses</td>
<td>2056</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_CutsFactor</td>
<td>2033</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_EachCutLimit</td>
<td>2102</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_GomoryCand</td>
<td>2048</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_GomoryPass</td>
<td>2050</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_Nodes</td>
<td>2017</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_PolishTime</td>
<td>2066</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_Populate</td>
<td>2108</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_ProbeDetTime</td>
<td>2150</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_ProbeTime</td>
<td>2065</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_RepairTries</td>
<td>2067</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_Solutions</td>
<td>2015</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_StrongCand</td>
<td>2045</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_StrongIt</td>
<td>2046</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_SubMIPNodeLim</td>
<td>2062</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Limits_TreeMemory</td>
<td>2027</td>
</tr>
<tr>
<td>CPXPARAM_MIP_OrderType</td>
<td>2032</td>
</tr>
<tr>
<td>CPXPARAM_MIP_PolishAfter_AbsMIPGap</td>
<td>2126</td>
</tr>
<tr>
<td>Constant Name</td>
<td>Value</td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Pool_AbsGap</td>
<td>2106</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Pool_Capacity</td>
<td>2103</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Pool_Intensity</td>
<td>2107</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Pool_Re1Gap</td>
<td>2105</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Pool_Replac</td>
<td>2104</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Backtrack</td>
<td>2002</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_BBInterval</td>
<td>2039</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Branch</td>
<td>2001</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_CallbackReducedLP</td>
<td>2055</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Dive</td>
<td>2060</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_File</td>
<td>2016</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_FPHeur</td>
<td>2098</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_HeristicFreq</td>
<td>2031</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_KappaStats</td>
<td>2137</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_LBHeur</td>
<td>2063</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_MIQCPStrat</td>
<td>2110</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_NodeSelect</td>
<td>2018</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Order</td>
<td>2020</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_PresolveNode</td>
<td>2037</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Probe</td>
<td>2042</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_RINSHeur</td>
<td>2061</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_Search</td>
<td>2109</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_StartAlgorithm</td>
<td>2025</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_SubAlgorithm</td>
<td>2026</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_VariableSelect</td>
<td>2028</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_AbsMIPGap</td>
<td>2008</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_Integrality</td>
<td>2010</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_LowerCutoff</td>
<td>2006</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_MIPGap</td>
<td>2009</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_ObjDiffrerence</td>
<td>2019</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_RelObjDifference</td>
<td>2022</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Tolerances_UpperCutoff</td>
<td>2007</td>
</tr>
<tr>
<td>CPXPARAM_Network_Display</td>
<td>5005</td>
</tr>
<tr>
<td>CPXPARAM_Network_Iterations</td>
<td>5001</td>
</tr>
<tr>
<td>CPXPARAM_Network_NetFind</td>
<td>1022</td>
</tr>
<tr>
<td>CPXPARAM_Network_Pricing</td>
<td>5004</td>
</tr>
<tr>
<td>CPXPARAM_Network_Tolerances_Feasibility</td>
<td>5003</td>
</tr>
<tr>
<td>CPXPARAM_Network_Tolerances_Optimality</td>
<td>5002</td>
</tr>
<tr>
<td>CPXPARAM_Output_CloneLog</td>
<td>1132</td>
</tr>
<tr>
<td>CPXPARAM_Output_IntSolFilePrefix</td>
<td>2143</td>
</tr>
<tr>
<td>CPXPARAM_Output_MPSLong</td>
<td>1081</td>
</tr>
<tr>
<td>CPXPARAM_Output_WriteLevel</td>
<td>1114</td>
</tr>
<tr>
<td>CPXPARAM_Parallel</td>
<td>1109</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Value</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Aggregator</td>
<td>1003</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_BoundStrength</td>
<td>2029</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_CoeffReduce</td>
<td>2004</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Dependency</td>
<td>1008</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Dual</td>
<td>1044</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Fill</td>
<td>1002</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Linear</td>
<td>1058</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_NumPass</td>
<td>1052</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Presolve</td>
<td>1030</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_QCPDuals</td>
<td>4003</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_QPMakePSD</td>
<td>4010</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Reduce</td>
<td>1057</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Relax</td>
<td>2034</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_RepeatPresolve</td>
<td>2064</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Symmetry</td>
<td>2059</td>
</tr>
<tr>
<td>CPXPARAM_QPMethod</td>
<td>1063</td>
</tr>
<tr>
<td>CPXPARAM_RandomSeed</td>
<td>1124</td>
</tr>
<tr>
<td>CPXPARAM_Read_APIEncoding</td>
<td>1130</td>
</tr>
<tr>
<td>CPXPARAM_Read_Constraints</td>
<td>1021</td>
</tr>
<tr>
<td>CPXPARAM_Read_DataCheck</td>
<td>1056</td>
</tr>
<tr>
<td>CPXPARAM_Read_FileEncoding</td>
<td>1129</td>
</tr>
<tr>
<td>CPXPARAM_Read_Nonzeros</td>
<td>1024</td>
</tr>
<tr>
<td>CPXPARAM_Read_QPNonzeros</td>
<td>4001</td>
</tr>
<tr>
<td>CPXPARAM_Read_Scale</td>
<td>1034</td>
</tr>
<tr>
<td>CPXPARAM_Read_Variables</td>
<td>1023</td>
</tr>
<tr>
<td>CPXPARAM_ScreenOutput</td>
<td>1035</td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Algorithm</td>
<td>1077</td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Display</td>
<td>1076</td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Iterations</td>
<td>1078</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Crash</td>
<td>1007</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_DGradient</td>
<td>1009</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Display</td>
<td>1019</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_Iterations</td>
<td>1020</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_LowerObj</td>
<td>1025</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_Perturbation</td>
<td>1028</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_Singularity</td>
<td>1037</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_UpperObj</td>
<td>1026</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Perturbation_Constant</td>
<td>1015</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Perturbation_Indicator</td>
<td>1027</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_PGradient</td>
<td>1029</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Pricing</td>
<td>1010</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Refactor</td>
<td>1031</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Feasibility</td>
<td>1016</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Markowitz</td>
<td>1013</td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Optimality</td>
<td>1014</td>
</tr>
<tr>
<td>CPXPARAM_SolutionTarget</td>
<td>1131</td>
</tr>
<tr>
<td>CPXPARAM_Threads</td>
<td>1067</td>
</tr>
<tr>
<td>CPXPARAM_TimeLimit</td>
<td>1039</td>
</tr>
</tbody>
</table>
new parameter names introduced in IBM ILOG CPLEX version 12.8.0

Callable library miscellaneous routines

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPXPARAM_Record</td>
<td>1162</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_SubMIPScale</td>
<td>2207</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_SubMIPStartAlg</td>
<td>2205</td>
</tr>
<tr>
<td>CPXPARAM_MIP_Strategy_SubMIPSubAlg</td>
<td>2206</td>
</tr>
<tr>
<td>CPXPARAM_ParamDisplay</td>
<td>1163</td>
</tr>
<tr>
<td>CPX_PARAM_PARAMDISPLAY</td>
<td>1163</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_ROW</td>
<td>1049</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_COL</td>
<td>1050</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_IND</td>
<td>1051</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_QLIN</td>
<td>1052</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_QUAD</td>
<td>1053</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF.LAZY</td>
<td>1054</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_UCUT</td>
<td>1055</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_RHS</td>
<td>1056</td>
</tr>
<tr>
<td>CPXMI_SAMECOEFF_OBJ</td>
<td>1057</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_CANDIDATE</td>
<td>0x0020</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_GLOBAL_PROGRESS</td>
<td>0x0010</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_LOCAL_PROGRESS</td>
<td>0x0008</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_RELAXATION</td>
<td>0x0040</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_THREAD_DOWN</td>
<td>0x0004</td>
</tr>
<tr>
<td>CPX_CALLBACKCONTEXT_THREAD_UP</td>
<td>0x0002</td>
</tr>
</tbody>
</table>

new parameter names introduced in IBM ILOG CPLEX version 12.9.0

Callable library miscellaneous routines

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPXPARAM_Read_WarningLimit</td>
<td>1157</td>
</tr>
<tr>
<td>CPXPARAM_MultiObjective_Display</td>
<td>1600</td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Folding</td>
<td>1164</td>
</tr>
</tbody>
</table>

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
cplexError-class

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

return_codeCPLEX, status_codeCPLEX, getParmValCPLEX

cplexError-class  Class "cplexError"

Description

Objects of class cpxerr are returned by various functions of cplexAPI, in order to distinguish a status (error) code from a successful result.

Objects from the Class

Objects can be created by calls of the form cplexError(err), with err being an error code of IBM ILOG CPLEX.

Slots

ernum: Object of class "integer" containing the error code.

Methods

err signature(object = "cplexError"): Prints an error message string corresponding to the error code.

ermsg signature(object = "cplexError"): Returns an error message string corresponding to the error code.

ernum signature(object = "cplexError"): Gets the error code.

ernum<- signature(object = "cplexError"): Sets the error code.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
cplexPtr-class

Description
Structure of the class "cplexPtr". Objects of that class are used to hold pointers to C structures used by IBM ILOG CPLEX.

Objects from the Class
Objects can be created by calls of the form
env <- openEnvCplex() and/or
prob <- initProbCplex(env)).

Slots

cplexPtrType: Object of class "character" giving the pointer type.
cplexPointer: Object of class "externalptr" containing the pointer to a C structure.

Methods

isCPLEXchanPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX channel, otherwise FALSE.
isCPLEXenvPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX environment, otherwise FALSE.
isCPLEXfilePointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX file, otherwise FALSE.
isCPLEXprobPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX problem object, otherwise FALSE.
isCPLEXtermPointer signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a pointer to a CPLEX termination signal, otherwise FALSE.
isNULLpointerCPLEX signature(object = "cplexPtr"): returns TRUE if cplexPointer(object) is a NULL pointer, otherwise FALSE.
cplexPointer signature(object = "cplexPtr"): gets the cplexPointer slot.
summary signature(object = "cplexPtr"): prints a summary of the problem object to the command line. If a solution is available, it prints also information retrieved by solutionCPLEX and solnInfoCPLEX. If no solution is available, it prints the corresponding error message. The method returns invisibly NULL. The CPLEX environment pointer is needed as second argument env to summary.
cplexPtrType signature(object = "cplexPtr"): gets the cplexPtrType slot.
cplexPtrType<- signature(object = "cplexPtr"): sets the cplexPtrType slot.
delColsCPLEX

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
openEnvCPLEX and initProbCPLEX

delColsCPLEX               Delete all Columns in a Specified Range

Description
Low level interface function to the IBM ILOG CPLEX function CPXdelcols. Consult the IBM
ILOG CPLEX documentation for more detailed information.

Usage
delColsCPLEX(env, lp, begin, end)

Arguments

<table>
<thead>
<tr>
<th>env</th>
<th>An object of class &quot;cplexPtr&quot; as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.</th>
</tr>
</thead>
<tbody>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
<tr>
<td>begin</td>
<td>Integer value, numeric index of the first column to be deleted.</td>
</tr>
<tr>
<td>end</td>
<td>Integer value, numeric index of the last column to be deleted.</td>
</tr>
</tbody>
</table>

Details
Interface to the C function delCols which calls the CPLEX function CPXdelcols.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

---

delFpDestCPLEX  Remove a File from the List of Message Destinations for a Channel

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelfpdest. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXdelfpdest has been removed.

Usage

delFpDestCPLEX(env, newch, cpfile)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
newch  A pointer to the channel for which destinations are to be deleted as returned by CPXaddchannel.
cpfile  Pointer to an IBM ILOG CPLEX file as returned by openFileCPLEX.

Details

Interface to the C function delFpDest which calls the CPLEX function CPXdelfpdest.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

addFpDestCPLEX
Delete a Range of Indicator Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelindconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

\texttt{delIndConstrsCplex(env, lp, begin, end)}

Arguments

- \texttt{env}: An object of class "cplexPtr" as returned by \texttt{openEnvCplex}. This is basically a pointer to an IBM ILOG CPLEX environment.
- \texttt{lp}: An object of class "cplexPtr" as returned by \texttt{initProbCplex}. This is basically a pointer to an IBM ILOG CPLEX problem object.
- \texttt{begin}: An integer that specifies the numeric index of the first indicator constraint to be deleted.
- \texttt{end}: An integer that specifies the numeric index of the last indicator constraint to be deleted.

Details

Interface to the C function \texttt{delIndConstrs} which calls the CPLEX function \texttt{CPXdelindconstrs}.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at \url{https://www.ibm.com/support/knowledgecenter/SSSA5P}. 
**Description**

Low level interface function to the IBM ILOG CPLEX function `CXPdelmipstarts`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```
delmipstartscplex(env, lp, begin, end)
```

**Arguments**

- **env**: An object of class `cplexPtr` as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class `cplexPtr` as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **begin**: An integer specifying the numeric index of the first MIP start to be deleted.
- **end**: An integer specifying the numeric index of the last MIP start to be deleted.

**Details**

Interface to the C function `delMIPstarts` which calls the CPLEX function `CXPdelmipstarts`.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

delNamesCPLEX

Remove all Names Assigned to Rows and Columns

Description

Low level interface function to the IBM ILOG CPLEX function cPXdelnames. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

delNamesCPLEX(env, lp)

Arguments

- **env**
  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

- **lp**
  An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function delNames which calls the CPLEX function CPLEXdelnames.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**Description**

Low level interface function to the IBM ILOG CPLEX function CPXfreetprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
delProbcplex(env, lp)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbcplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function delProb which calls the CPLEX function CPXfreetprob.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

**See Also**

`initProbcplex`
**delQConstrsCPLEX**

**Delete a Range of Quadratic Constraints**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXdelqconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
delQConstrsCPLEX(env, lp, begin, end)
```

**Arguments**

- **env**
  - An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- **lp**
  - An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

- **begin**
  - An integer that specifies the numeric index of the first quadratic constraint to be deleted.

- **end**
  - An integer that specifies the numeric index of the last quadratic constraint to be deleted.

**Details**

Interface to the C function `delQConstrs` which calls the CPLEX function CPXdelqconstrs.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

delRowsCplex(env, lp, begin, end)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin
Integer value, numeric index of the first row to be deleted.

end
Integer value, numeric index of the last row to be deleted.

Details

Interface to the C function delRows which calls the CPLEX function CPXdelrows.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Delete a Set of Columns

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelsetcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

delsetcolscplex(env, lp, delstat)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

delstat An array specifying the columns to be deleted.

Details

Interface to the C function delSetCols which calls the CPLEX function CPXdelsetcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliedie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Delete a Set of Rows

Description

Low level interface function to the IBM ILOG CPLEX function CPXdelsetrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
delSetRowsCPLEX(env, lp, delstat)
```

Arguments

- `env`: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `delstat`: An array specifying the rows to be deleted.

Details

Interface to the C function delSetRows which calls the CPLEX function CPXdelsetrows.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldor6.de>

Maintainer: Mayo Roettger <mayo.roettger@hu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**delTerminateCPLEX**  
*Terminate CPLEX gracefully*

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxsetterminate`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
delTerminateCPLEX(env, tsig)
```

**Arguments**

- `env`  
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- `tsig`  
  Pointer to termination signal as returned by `setTerminateCPLEX`.

**Details**

Interface to the C function `setTerminate` which calls the CPLEX function `CplexSetterminate`.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>  
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`setTerminateCPLEX`, `printTerminateCPLEX`, `chgTerminateCPLEX`
disconnectChannelCplex

Flush all Message Destinations Associated with a Channel

Description

Low level interface function to the IBM ILOG CPLEX function CPXdisconnectchannel. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

disconnectChannelCplex(env, newch)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- newch: A pointer to the channel containing the message destinations as returned by CPXaddChannel.

Details

Interface to the C function disconnectChannel which calls the CPLEX function CPXdisconnectchannel.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

flushChannelCplex, flushStdChannelsCplex, getChannelsCplex
Find a Problem Solution Using the Dual Simplex Algorithm

Description

Low level interface function to the IBM ILOG CPLEX function CPXdualopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

\[ \text{dualoptCLEX}(\text{env, lp}) \]

Arguments

- \( \text{env} \): An object of class "cplexPtr" as returned by \texttt{openEnvCPLEX}. This is basically a pointer to an IBM ILOG CPLEX environment.
- \( \text{lp} \): An object of class "cplexPtr" as returned by \texttt{initProbCLEX}. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function dualopt which calls the CPLEX function CPXdualopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

\texttt{solnInfoCPLEX, getStatCLEX, solutionCLEX}
Write a Dual Formulation of the Current CPLEX Problem Object

dualWriteCplex

Description

Low level interface function to the IBM ILOG CPLEX function CPXdualwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

dualWriteCplex(env, lp, fname)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

fname Single character value giving the filename to write to.

Details

Interface to the C function dualWrite which calls the CPLEX function CPXdualwrite.

Value

Zero if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
feasOptCPLEX

*Compute a Minimum-Cost Relaxation*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXfeasopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
desoptcplex(env, lp, rhs = FALSE, rng = FALSE, lb = FALSE, ub = FALSE)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `rhs` If set to FALSE no right hand side value is allowed to be relaxed.
- `rng` If set to FALSE no range values are allowed to be relaxed.
- `lb` If set to FALSE no lower bound of any variable is allowed to be relaxed.
- `ub` If set to FALSE no lower bound of any variable is allowed to be relaxed.

**Details**

Interface to the C function `feasOpt` which calls the CPLEX function CPXfeasopt.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`solutionCPLEX, getRowInfeasCPLEX, getColInfeasCPLEX, solnInfoCPLEX, getStatCPLEX`
fileputCPLEX  Write to File

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxfputs`. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where `cpxfputs` has been removed.

**Usage**

```r
cfileputcplex(cpfile, stuff = "")
```

**Arguments**

- *cpfile*: A pointer to a file as returned by `openFileCPLEX`.
- *stuff*: A character string to be written to the file.

**Details**

Interface to the C function `fileput` which calls the CPLEX function `Cpxfputs`.

**Value**

A nonnegative value if successful, otherwise EOF.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

**See Also**

`closeFileCPLEX`, `openFileCPLEX`
flushChannelCPLEX              Flush All Message Destinations Associated With a Channel

Description

Low level interface function to the IBM ILOG CPLEX function CPXflushchannel. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

flushChannelCPLEX(env, newch)

Arguments

denv                  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
dnewch                Pointer to a channel object as returned by addChannelCPLEX.

Details

Interface to the C function flushChannel which calls the CPLEX function CPXflushchannel.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

disconnectChannelCPLEX, flushStdChannelsCPLEX, getChannelsCPLEX
flushStdChannelsCPLEX  
*Flushes the Output Buffers of the Four Standard Channels*

**Description**

Low level interface function to the IBM ILOG CPLEX function `CpXflushstdchannels`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

`flushStdChannelsCPLEX(env)`

**Arguments**

`env`  
An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

**Details**

Interface to the C function `flushStdChannels` which calls the CPLEX function `CpXflushstdchannels`.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich  
geliudie@uni-duesseldorf.de  
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`disconnectChannelCPLEX`, `flushChannelCPLEX`, `getChannelsCPLEX`
Description

Low level interface function to the IBM ILOG CPLEX function CPXfreepresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

freepresolveCPLEX(env, lp)

Arguments

env          An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp           An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function freePresolve which calls the CPLEX function CPXfreepresolve.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getBaseCPLEX

Access Basis Resident in a CPLEX Problem Object.

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetbase. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getBaseCPLEX(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getBase which calls the CPLEX function CPXgetbase.

Value

If successful a list is returned:

cstat basis status of the columns in the CPLEX problem object

rstat basis status of the artificial, slack, or surplus variable associated with each row in the constraint matrix

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getBestObjValCPLEX  

Access the Currently Best Known Bound of all the Remaining Open Nodes in a Branch-And-Cut Tree

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetbestobjval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getBestObjValCPLEX(env, lp)

Arguments

env  
An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getBestObjVal which calls the CPLEX function CPXgetbestobjval.

Value

Objective value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getChannelsCPLEX

Obtain Pointers to the Four Default Channels

Description
Low level interface function to the IBM ILOG CPLEX function CPLEX::getchannels. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getChannelsCPLEX(env, ptrtype = "cplex_chan")

Arguments
env          An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
ptrtype      A name for the pointer object.

Details
Interface to the C function getChannels which calls the CPLEX function CPLEX::getchannels.

Value
If successful a list is returned:
cpxresults  address of the channel corresponding to cpxresults
cpxwarning  address of the channel corresponding to cpxwarning
cpxerror    address of the channel corresponding to cpxerror
cpxlog       address of the channel corresponding to cpxlog
otherwise an instance of class "cplexError". Each list element is an object of class "cplexPtr".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
disconnectChannelCPLEX, flushChannelCPLEX, flushStdChannelsCPLEX
getChgParmCplex

Get Parameter Numbers for Parameters Which are Not Set at Their Default Values

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetchgparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getChgParmCplex(env)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getChgParm which calls the CPLEX function CPXgetchgparam.

Value

A vector containing integer values (unique parameter identifiers) for parameters which are not set at their default values, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgcenter/SSSA5P.

See Also

cplexConstants
getCoefCPLEX

Access a Single Constraint Matrix Coefficient

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcoef. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getCoefCPLEX(env, lp, i, j)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- i: An integer specifying the numeric index of the row.
- j: An integer specifying the numeric index of the column.

Details

Interface to the C function getCoef which calls the CPLEX function CPXgetcoef.

Value

Matrix coefficient value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldor.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getColIndexCPLEX

Search for the Index Number of the Specified Column

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getColIndexCPLEX(env, lp, cname)

Arguments

- **env**: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **cname**: A column name to search for.

Details

Interface to the C function `getColIndex` which calls the CPLEX function CPXgetcolindex.

Value

Column number if successful, otherwise an instance of class "cplexError".

Author(s)

- Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

getColInfeasCPLEX  
Compute Infeasibility of a Given Solution for a Range of Variables

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolinfeas. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```c
getColInfeasCPLEX(env, lp, begin, end, sol = NULL)
```

Arguments

- `env` An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin` An integer specifying the beginning of the range of variables whose infeasibility is to be returned.
- `end` An integer specifying the end of the range of variables whose infeasibility is to be returned.
- `sol` The solution whose infeasibility is to be computed.

Details

Interface to the C function getColInfeas which calls the CPLEX function CPXgetcolinfeas.

Value

Infeasibility values if successful, otherwise an instance of class "cplexError".

Author(s)

- Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getColNameCplex

Access a Range of Column Names

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcolname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getColNameCplex(env, lp, begin, end)

Arguments

env               An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp                An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
begin             An integer specifying the beginning of the range of column names to be returned.
end               An integer specifying the end of the range of column names to be returned.

Details

Interface to the C function getColName which calls the CPLEX function CPXgetcolname.

Value

Column names if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getColsCPLEX \hspace{1cm} \textit{Accesses a Range of Columns of the Constraint Matrix}

\textbf{Description}

Low level interface function to the IBM ILOG CPLEX function CPXgetcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

\textbf{Usage}

\begin{verbatim}
getColsCPLEX(env, lp, begin, end)
\end{verbatim}

\textbf{Arguments}

\begin{itemize}
\item \texttt{env} \hspace{1cm} An object of class "\texttt{cplexPtr}" as returned by \texttt{openEnvCplex}. This is basically a pointer to an IBM ILOG CPLEX environment.
\item \texttt{lp} \hspace{1cm} An object of class "\texttt{cplexPtr}" as returned by \texttt{initProbCplex}. This is basically a pointer to an IBM ILOG CPLEX problem object.
\item \texttt{begin} \hspace{1cm} An integer specifying the beginning of the range of columns to be returned.
\item \texttt{end} \hspace{1cm} An integer specifying the end of the range of columns to be returned.
\end{itemize}

\textbf{Details}

Interface to the C function \texttt{getCols} which calls the CPLEX function CPXgetcols.

\textbf{Value}

If successful a list is returned:

\begin{itemize}
\item \texttt{matbeg} \hspace{1cm} Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
\item \texttt{matind} \hspace{1cm} Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
\item \texttt{matval} \hspace{1cm} Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
\end{itemize}

otherwise an instance of class "\texttt{cplexError}".

\textbf{Author(s)}

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

\textbf{References}

The IBM ILOG CPLEX home page at \url{https://www.ibm.com/support/knowledgecenter/SSSA5P}. 
getColTypeCPLEX

Access Types for a Range of Variables

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetctype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getColTypeCPLEX(env, lp, begin, end)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
begin An integer specifying the beginning of the range of the types to be returned.
end An integer specifying the end of the range of the types to be returned.

Details

Interface to the C function getColType which calls the CPLEX function CPXgetctype.

Value

Column types if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getConflictCplex

Return Linear Constraints and Variables Belonging to a Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetconflict. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getConflictCplex(env, lp)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getConflict which calls the CPLEX function CPXgetconflict.

Value

If successful a list is returned:

confstat    status of the conflict
confnumrows number of rows in the conflict
rowind      indices of the constraints that participate in the conflict
rowbdstat   conflict status of the rows
confnumcols number of columns in the conflict
colind      indices of the variables that participate in the conflict
colbdstat   conflict status of the columns

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getConflictExtCplex

See Also
cplexConstants

---

getConflictExtCPLEX  Get Conflict Status Codes

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetconflictext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getConflictExtCPLEX(env, lp, begin, end)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
<tr>
<td>begin</td>
<td>The index of the first group.</td>
</tr>
<tr>
<td>end</td>
<td>The index of the last group.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getConflictExt which calls the CPLEX function CPXgetconflictext.

Value

Specified values denoting the conflict status if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getCutoffCPLEX

Access MIP Cutoff Value Being Used During Mixed Integer Optimization.

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetcutoff. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getCutoffCPLEX(env, lp)

Arguments

- `env`: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

- `lp`: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getCutoff which calls the CPLEX function CPXgetcutoff.

Value

Value of the cutoff if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getDblParmCplex**

*Obtain the Current Value of a CPLEX Parameter of Type Double*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXgetdblparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

`getDblParmCplex(env, parm)`

**Arguments**

- **env**: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **parm**: Constant or reference number of the desired parameter.

**Details**

Interface to the C function `getdblParm` which calls the CPLEX function CPXgetdblparam.

**Value**

Parameter value if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

cplexConstants
getDb1QualCPLEX  

Get Double-Valued Information About the Quality of the Current Solution of a Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdblquality. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getDb1QualCPLEX(env, lp, w)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **w**: An Integer specifying the quality value to be retrieved.

Details

Interface to the C function getDb1Qual which calls the CPLEX function CPXgetdblquality.

Value

Requested quality value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getDbsCntCPLEX

Access the Number of Dual Super-Basic Variables in the Current Solution

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdsbcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getDbsCntCPLEX(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getDbsCnt which calls the CPLEX function CPXgetdsbcnt.

Value

Number of dual super-basic variables if a solution exists, otherwise zero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getDjCPLEX

Accesses Reduced Costs for a Range of Variables of a Linear or Quadratic Program

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetdj. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getAddressCPLEX(env, lp, begin, end)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- begin: An integer specifying the beginning of the range of reduced-cost values to be returned.
- end: An integer specifying the end of the range of reduced-costs values to be returned.

Details

Interface to the C function getDj which calls the CPLEX function CPXgetdj.

Value

Reduced costs if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgcenter/SSSA5P.
getErrorStrCplex

Return an Error Message String Corresponding to an Error Code

Description

Low level interface function to the IBM ILOG CPLEX function CPXgeterrorstring. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

ggetErrorStrCplex(err, env = NULL)

Arguments

err The error code to be translated.
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getErrorStr which calls the CPLEX function CPXgeterrorstring.

Value

A single character value containing the error message string.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

ggetOptionCplex
getGradCplex

Project the Impact of Making Changes to Optimal Variable Values or Objective Function Coefficients

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetgrad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getGradCplex(env, lp, j)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
j An integer specifying the index of the column of interest.

Details

Interface to the C function getGrad which calls the CPLEX function CPXgetgrad.

Value

If successful a list is returned:

head listing of the indices of the basic variables in the order in which they appear in the basis.
y coefficients of the j-th column relative to the current basis.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getIndConstrCplex**

Access a Specified Indicator Constraint on the Variables of a CPLEX Problem Object.

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetindconstr`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getIndConstrCplex(env, lp, which)
```

**Arguments**

- `env`: An object of class "`cplexPtr`" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "`cplexPtr`" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `which`: An integer specifying which indicator constraint to return.

**Details**

Interface to the C function `getIndConstr` which calls the CPLEX function `CPXgetindconstr`.

**Value**

If successful a list is returned:

- `indvar`: Index of the binary indicator variable. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `complemented`: Boolean value that specifies whether the indicator variable is complemented. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `rhs`: Righthand side value of the linear portion of the indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `sense`: Sense of the linear portion of the constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `linind`: Variable indices of the entries of `linval`. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `linval`: Coefficients of the linear portion of the specified indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "`cplexError`".
getInfoDblParmCplex

**Author(s)**
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

---

**getInfoDblParmCplex**  
*Obtain Default, Minimum and Maximum Values of a Parameter of Type Double*

**Description**
Low level interface function to the IBM ILOG CPLEX function `CpXinfodblparam`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**
```
getInfoDblParmCplex(env, parm)
```

**Arguments**
- **env**  
  An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **parm**  
  Constant or reference number of the desired parameter.

**Details**
Interface to the C function `getInfoDblParm` which calls the CPLEX function `CpXinfodblparam`.

**Value**
If successful a list is returned:
- **defvalue**  
  default value
- **minvalue**  
  minimum value
- **maxvalue**  
  maximum value

otherwise an instance of class "cplexError".

**Author(s)**
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants

getInfoIntParmCPLEX | Obtain Default, Minimum and Maximum Values of a Parameter of Type CPXINT

description

Low level interface function to the IBM ILOG CPLEX function CPXinfointparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getInfoIntParmCPLEX(env, parm)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
parm Constant or reference number of the desired parameter.

details

Interface to the C function getinfoIntParm which calls the CPLEX function CPXinfointparam.

Value

If successful a list is returned:

defvalue default value
minvalue minimum value
maxvalue maximum value

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getInfoLongParmCPLEX**  
*Obtain Default, Minimum and Maximum Values of a Parameter of Type CPXLONG*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXinfolongparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getInfoLongParmCLEX(env, parm)
```

**Arguments**

- `env`  
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- `parm`  
  Constant or reference number of the desired parameter.

**Details**

Interface to the C function `getInfoLongParm` which calls the CPLEX function CPXinfolongparam.

**Value**

If successful a list is returned:

- `defvalue`  
  default value

- `minvalue`  
  minimum value

- `maxvalue`  
  maximum value

otherwise an instance of class "cplexError".

**Note**

In order to get a 64 bit integer value from CPXinfolongparam, datatype numeric is used. All return values will be numeric.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
getInfoStrParmCPLEX

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
g得以ntParmCPLEX, cplexConstants

getInfoStrParmCPLEX  

Obtain Default Value of a String Parameter

Description
Low level interface function to the IBM ILOG CPLEX function CPXinfostrparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
g得以InfoStrParmCPLEX(env, parm)

Arguments
env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
parm Constant or reference number of the desired parameter.

Details
Interface to the C function getInfoStrParm which calls the CPLEX function CPXinfostrparam.

Value
A single character value.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants
### getIntParmCPLEX

| **Obtain the Current Value of a CPLEX Parameter of Type CPXINT** |

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxgetintparam`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getIntParmCPLEX(env, parm)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `parm` Constant or reference number of the desired parameter.

**Details**

Interface to the C function `getIntParm` which calls the CPLEX function `CPXgetintparam`.

**Value**

Parameter value if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

**See Also**

`cplexConstants`
getIntQualCPLEX

Access Integer-Valued Information About the Quality of the Current Solution of a Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetintquality. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getIntQualCPLEX(env, lp, w)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

w An Integer specifying the quality value to be retrieved.

Details

Interface to the C function getIntQual which calls the CPLEX function CPXgetintquality.

Value

Requested quality value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getItCntCPLEX

Access the Total Number of Simplex Iterations to Solve an LP Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetitcnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getItCntCPLEX(env, lp)

Arguments

env  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
lp   An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function `getItCnt` which calls the CPLEX function CPXgetitcnt.

Value

Total iteration count if solution exists, otherwise zero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getLogFileCplex

Access log file to Which Messages are Written

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXgetlogfile has been removed.

Usage

getLogFileCplex(env, ptrtype = "cplex_file")

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

ptrtype
A name for the pointer object.

Details

Interface to the C function getLogFile which calls the CPLEX function CPXgetlogfile.

Value

If successful, a pointer to the CPLEX file is returned (an instance of class "cplexPtr"), otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgcenter/SSSA5P.

See Also

setLogFileCplex
getLogFileNameCLEX

Get the name of the current logfile

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetlogfilename. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.8.0, where CPXgetlogfilename was not included.

Usage

getLogFileNameCLEX(env)

Arguments

env

An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getLogFile which calls the CPLEX function CPXgetlogfilename.

Value

Zero if successful, otherwise nonzero.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
**getLongParmCPLEX**

*Obtain Current Value of a Parameter of Type CPXLONG*

---

**Description**

Low level interface function to the IBM ILOG CPLEX function CPLEXgetlongparm. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

`getLongParmCPLEX(env, parm)`

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `parm` Constant or reference number of the desired parameter.

**Details**

Interface to the C function `getLongParm` which calls the CPLEX function CPLEXgetlongparm.

**Value**

Parameter value if successful, otherwise an instance of class "cplexError".

**Note**

In order to get a 64 bit integer value from CPLEXgetlongparm, datatype numeric is used. The return value will be numeric.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`getIntParmCPLEX`, `cplexConstants`
getLowBndsIdsCplex

Retrieve Lower Bounds on Variables

Description

The function retrieves the lower bounds on specified variables.

Usage

getLowBndsIdsCplex(env, lp, ind)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
ind Column indices of variables (remember: first index is 0).

Value

A numeric vector containing the lower bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getLowerBndsCplex
**getLowerBndsCPLEX**

*Access a Range of Lower Bounds on Variables*

**Description**

Low level interface function to the IBM ILOG CPLEX function `CpxGetlb`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getLowerBndsCPLEX(env, lp, begin, end)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin` Beginning of the range of lower bounds to be returned.
- `end` End of the range of lower bounds to be returned.

**Details**

Interface to the C function `getLowerBnds` which calls the CPLEX function `CpxGetlb`.

**Value**

A numeric vector containing the lower bounds on the specified variables. If not successfull an instance of class "cplexError" is returned.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <maya.roettger@hhu.de>

**References**

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getMethodCPLEX

Obtain Solution Algorithm

Description
Low level interface function to the IBM ILOG CPLEX function cpxgetmethod. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getMethodCPLEX(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function getMethod which calls the CPLEX function CPXgetmethod.

Value
A single integer value specifying the solution algorithm.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants section “LP/QP solution algorithms”.
getMIPrelGapCPLEX

Access Relative Objective Gap for a MIP Optimization

Description
Low level interface function to the IBM ILOG CPLEX function CPXgetmiprelgap. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getMIPrelGapCPLEX(env, lp)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function getMIPrelGap which calls the CPLEX function CPXgetmiprelgap.

Value
Relative Objective Gap if successful, otherwise an instance of class "cplexError".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
getObjValCplex, getBestObjValCplex
getMIPstartIndexCPLEX  Search for the Index Number of the Specified MIP Start

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmipstartindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getMIPstartIndexCPLEX(env, lp, iname)

Arguments

env      An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp       An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
iname    A MIP start name to search for.

Details

Interface to the C function getMIPstartIndex which calls the CPLEX function CPXgetmipstartindex.

Value

Index number of the specified MIP start if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getMIPstartNameCPLEX  Access a Range of Names of MIP Starts

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetmipstartname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getMIPstartNameCPLEX(env, lp, begin, end)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp  An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
begin  An integer specifying the beginning of the range of MIP starts to be returned.
end  An integer specifying the end of the range of MIP starts to be returned.

Details

Interface to the C function getMIPstartName which calls the CPLEX function CPXgetmipstartname.

Value

Names of the MIP starts if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich < geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger < mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getMIPstartsCplex**

*Access a Range of MIP Starts of a CPLEX Problem Object*

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxgetmipstarts`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getMIPstartsCplex(env, lp, begin, end)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin` An integer specifying the beginning of the range of MIP starts to be returned.
- `end` An integer specifying the end of the range of MIP starts to be returned.

**Details**

Interface to the C function `getMIPstarts` which calls the CPLEX function `cpxgetmipstarts`.

**Value**

If successful a list is returned:

- `beg` Array specifying where each of the requested MIP starts begins in the arrays `varindices` and `values`. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `varindices` Array containing the numeric indices of the columns corresponding to the variables which are assigned starting values. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `values` Array containing the values of the MIP starts. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `effortlevel` Array containing the effort level for each MIP start requested. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelsius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
**getNumColsCPLEX**

**Access the Number of Columns in the Constraint Matrix**

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetnumcols`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getNumColsCPLEX(env, lp)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function `getNumCols` which calls the CPLEX function `CPXgetnumcols`.

**Value**

If successful the number of variables is returned. If `env` or `lp` do not exist, zero is returned.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

Access the Number of MIP Starts in the CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnummipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getNumMIPstartsCplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getNumMIPstarts which calls the CPLEX function CPXgetnummipstarts.

Value

If successful the number of MIP starts is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getNumNnzCplex

Access the Number of Nonzero Elements in the Constraint Matrix

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumnz. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

ggetNumNnzCplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getNumNnz which calls the CPLEX function CPXgetnumnz.

Value

Zero if the problem object or environment does not exist, otherwise the number of nonzero elements.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getNumQConstrsCPELEX  Return the Number of quadratic constraints.

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumqconstrs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getNumQConstrsCPELEX(env, lp)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCPELEX. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCPELEX. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getNumQConstrs which calls the CPLEX function CPXgetnumqconstrs.

Value

If successful the number of quadratic constraints is returned. If env or lp do not exist, zero is returned.

Author(s)

Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getNumQPnzCplex

Return the Number of Nonzeros in the $Q$ Matrix

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumqpnz. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

ggetNumQPnzCplex(env, lp)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getNumQPnz which calls the CPLEX function CPXgetnumqpnz.

Value

If successful the number of nonzeros in the $Q$ matrix is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getNumQuadCplex

Return the Number of Variables That Have Quadratic Objective Coefficients

Description
Low level interface function to the IBM ILOG CPLEX function CPXgetnumquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getNumQuadCplex(env, lp)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function getNumQuad which calls the CPLEX function CPXgetnumquad.

Value
If successful the number of variables that have quadratic objective coefficients is returned. If env or lp do not exist, zero is returned.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getNumRowsCPLex

Access the Number of Rows in the Constraint Matrix

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetnumcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getNumRowsCPLex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getNumRows which calls the CPLEX function CPXgetnumrows.

Value

If successful the number of rows is returned. If env or lp do not exist, zero is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSSA5P.
getObjCPLEX

Access a Range of Objective Function Coefficients of a CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobj. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getobjcplex(env, lp, begin, end)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
begin An integer specifying the beginning of the range of objective function coefficients to be returned
end An integer specifying the end of the range of objective function coefficients to be returned.

Details

Interface to the C function getobj which calls the CPLEX function CPXgetobj.

Value

Specified objective coefficients if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getObjDirCPLEX

Access the Direction of Optimization

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetobjsen`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getObjDirCPLEX(env, lp)
```

**Arguments**

- `env`: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function `getobjDir` which calls the CPLEX function `CPXgetobjsen`.

**Value**

Zero if the problem object or environment does not exist, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

cplexConstants
Description
Low level interface function to the IBM ILOG CPLEX function CPXgetobjname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getobjNameCplex(env, lp)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function getObjName which calls the CPLEX function CPXgetobjname.

Value
Name of the objective row if successful, otherwise an instance of class "cplexError".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getObjOffsetCplex

Objective Offset Between the Original Problem and the Presolved Problem.

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjoffset. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getObjOffsetCplex(env, lp)

Arguments

- `env` An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getobjoffset which calls the CPLEX function CPXgetobjoffset.

Value

Objective offset value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getObjValCplex

Access Solution Objective Value

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetobjval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getObjValCplex(env, lp)

Arguments

env  
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getObjVal which calls the CPLEX function CPXgetobjval.

Value

Objective value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getOrderCPLEX  

Access MIP Priority Order Information

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetorder. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getOrderCPLEX(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getorder which calls the CPLEX function CPXgetorder.

Value

If successful a list is returned:

indices indices of the variables in the order
priority priority values
direction preferred branching directions

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getParmHierNameCplex: Obtain the hierarchy name string of a CPLEX parameter

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamhiername. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.9.0 on, where CPXgetparamhiername was not included.

Usage

getParmHierNameCplex(env, whichparam)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

whichparam An integer specifying the symbolic constant (or reference number) of the desired parameter.

Details

Interface to the C function getParmHierName which calls the CPLEX function CPXgetparamhiername.

Value

A single character value.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getParmNameCplex

Obtain the Name of a CPLEX Parameter, Given the Symbolic Constant

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getParmNameCplex(env, wparm)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

wparm Constant or reference number of the desired parameter.

Details

Interface to the C function getParmName which calls the CPLEX function CPXgetparamname.

Value

A single character value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getParmNumCPLEX

Obtain the Reference Number of a CPLEX Parameter

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamnum. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getParmNumCPLEX(env, nparm)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

nparm A single character value containing the name of the parameter.

Details

Interface to the C function getParmNum which calls the CPLEX function CPXgetparamnum.

Value

A single integer value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getParmTypeCplex

Obtain the Type of a CPLEX Parameter

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetparamtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getParmTypeCplex(env, parm)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
parm Constant or reference number of the desired parameter.

Details

Interface to the C function getParmType which calls the CPLEX function CPXgetparamtype.

Value

A single integer value if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getParmValCplex Values and Names of Parameters Having Non-Default Values

Description

The function getParmValCplex retrieves the names and actual values of all IBM ILOG CPLEX parameters, which do not have their default values.

Usage

getParmValCplex(env)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

Value

Either a list containing all non-default parameters and their values or NULL.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants and getChgParmCplex

getPhase1CntCplex Access Number of Phase I Iterations

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetPhase1cnt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getPhase1CntCplex(env, lp)
getPiCPLEX

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getPhase1Cnt which calls the CPLEX function CPXgetphase1cnt.

Value

Zero if no solution exists, otherwise Phase I iteration count.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

getiCPLEX

Access Dual Values for a Range of Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetpi. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getiCLEX(env, lp, begin, end)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin An integer specifying the beginning of the range of dual values to be returned.

end An integer specifying the end of the range of dual values to be returned.

Details

Interface to the C function getPi which calls the CPLEX function CPXgetpi.
getPreStatCplex

Value

Values of the dual variables if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

---

getPreStatCplex Access Presolve Status Information for Columns and Rows

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetprestat. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getPreStatCplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getPreStat which calls the CPLEX function CPXgetprestat.

Value

If successful a list is returned:

prestat status of the presolved problem
pcstat presolve status values of the columns
prstat presolve status values of the rows
ocstat presolve status values of the columns of the presolved problem
orstat presolve status values of the rows of the presolved problem

otherwise an instance of class "cplexError".
getProbNameCplex

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants

getProbNameCplex Access Problem Name

Description
Low level interface function to the IBM ILOG CPLEX function CPXgetprobname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getProbNameCplex(env, lp)

Arguments

<table>
<thead>
<tr>
<th>env</th>
<th>An object of class &quot;cplexPtr&quot; as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.</th>
</tr>
</thead>
<tbody>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
</tbody>
</table>

Details
Interface to the C function getProbName which calls the CPLEX function CPXgetprobname.

Value
Name of the problem if successful, otherwise an instance of class "cplexError".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getProbTypeCplex

Access Problem Type

Description
Low level interface function to the IBM ILOG CPLEX function CPXgetprobtype. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getProbTypeCplex(env, lp)

Arguments
- env: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function getProbType which calls the CPLEX function CPXgetprobtype.

Value
A single integer value specifying the problem type.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
chgProbTypeCplex, cplexConstants section “Problem Types”.
getProbVarCPLEX

Access the Solution Values for a Range of Problem Variables

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetx. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getProbVarCPLEX(env, lp, begin, end)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin An integer specifying the beginning of the range of variable values to be returned.

dend An integer specifying the end of the range of variable values to be returned.

Details

Interface to the C function getProbVar which calls the CPLEX function CPXgetx.

Value

Values of the primal variables if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getQConstrCplex

Access a Specified Quadratic Constraint on the Variables of a CPLEX Problem Object.

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXgetqconstr. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getQConstrCplex(env, lp, which)
```

**Arguments**

- **env**: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **which**: An integer specifying which quadratic constraint to return.

**Details**

Interface to the C function `getQConstr` which calls the CPLEX function CPXgetqconstr.

**Value**

If successful a list is returned:

- **rhs**: Righthand-side value of the quadratic constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **sense**: Character specifying the sense of the constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **linind**: Variable indices of the entries of `linval`. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **linval**: Linear coefficients of the specified constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **quadrow**: Variable indices of the entries of `quadval`. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **quadcol**: Variable indices of the entries of `quadval`. Consult the IBM ILOG CPLEX documentation for more detailed information.
- **quadval**: Quadratic coefficients of the specified constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".
**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxgetqpcoef`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getQPcoefCplex(env, lp, i, j)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `i` The row number in Q.
- `j` The row column in Q.

**Details**

Interface to the C function `getQPcoef` which calls the CPLEX function `CPXgetqpcosf`.

**Value**

Specified quadratic coefficient in the matrix Q if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

getQuadCPELEX

Access a Range of Columns of the Matrix $Q$ of a Model With a Quadratic Objective Function

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetquad. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getQuadCPELEX(env, lp, begin, end)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPELEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPELEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin
An integer specifying the beginning of the range of columns to be returned.

dend
An integer specifying the end of the range of columns to be returned.

Details

Interface to the C function getQuad which calls the CPLEX function CPXgetquad.

Value

If successful a list is returned:

qmatbeg
Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.

qmatind
Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.

qmatval
Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getRedLpCplex

Get a Pointer for the Presolved Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetredlp. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getRedLpCplex(env, lp, ptrtype = "cplex_prob")

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

ptrtype
A name for the pointer object.

Details

Interface to the C function getRedLp which calls the CPLEX function CPXgetredlp.

Value

Pointer for the presolved problem if successful (an instance of class "cplexPtr"), otherwise an instance of class "cplexError" or NULL.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getRhsCplex

Access Righthand Side Coefficients for a Range of Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrhs. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getRhsCplex(env, lp, begin, end)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>env</td>
<td>An object of class &quot;cplexPtr&quot; as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.</td>
</tr>
<tr>
<td>lp</td>
<td>An object of class &quot;cplexPtr&quot; as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
<tr>
<td>begin</td>
<td>An integer specifying the beginning of the range of righthand side terms to be returned.</td>
</tr>
<tr>
<td>end</td>
<td>An integer specifying the end of the range of righthand side terms to be returned.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getRhs which calls the CPLEX function CPXgetrhs.

Value

Specified righthand side coefficients if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getRngValCPLEX

Accesses Righthand Side Coefficients Range Coefficients

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrngval. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getRngValCPLEX(env, lp, begin, end)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **begin**: An integer specifying the beginning of the set of rows for which RHS range coefficients are returned.
- **end**: An integer specifying the end of the set of rows for which RHS range coefficients are returned.

Details

Interface to the C function getRngVal which calls the CPLEX function CPXgetrngval.

Value

Specified RHS range coefficients if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getRowIndexCPLEX

Search for the Index Number of a Specified Row

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrowindex. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getiRowIndexCPLEX(env, lp, rname)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **rname**: A row name to search for.

Details

Interface to the C function getRowIndex which calls the CPLEX function CPXgetrowindex.

Value

Specified row index if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getRowInfeasCplex**

**Compute Infeasibility of a Given Solution for a Range of Linear Constraints**

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetrowinfeas`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
getRowInfeasCplex(env, lp, begin, end, sol = NULL)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin` An integer specifying the beginning of the range of linear constraints whose infeasibility is to be returned.
- `end` An integer specifying the beginning of the range of linear constraints whose infeasibility is to be returned.
- `sol` The solution whose infeasibility is to be computed.

**Details**

Interface to the C function `getRowInfeas` which calls the CPLEX function `CPXgetrowinfeas`.

**Value**

Infeasibility values if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

getRowNameCPLEX  

Access a Range of Row Names

Description
Low level interface function to the IBM ILOG CPLEX function CPXgetrownname. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
getRowNameCPLEX(env, lp, begin, end)

Arguments
- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- begin: An integer specifying the beginning of the range of row names to be returned.
- end: An integer specifying the end of the range of row names to be returned.

Details
Interface to the C function getRowName which calls the CPLEX function CPXgetrowname.

Value
Specified row names if successful, otherwise an instance of class "cplexError".

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getRowsCPLEX**

*Accesses a Range of Rows of the Constraint Matrix*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXgetrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getRowsCPLEX(env, lp, begin, end)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin` An integer specifying the beginning of the range of rows to be returned.
- `end` An integer specifying the end of the range of rows to be returned.

**Details**

Interface to the C function `getRows` which calls the CPLEX function CPXgetrows.

**Value**

If successful a list is returned:

- `matbeg` Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matind` Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.
- `matval` Array that specifies the nonzero elements of the rows. Consult the IBM ILOG CPLEX documentation for more detailed information.

otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

getSenseCplex  

Access the Sense for a Range of Constraints in a CPLEX Problem Object.

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsense. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getsensecplexHenvL lpL beginL endI

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin An integer specifying the beginning of the range of constraint senses to be returned.

der An integer specifying the end of the range of constraint senses to be returned.

Details

Interface to the C function getSense which calls the CPLEX function CPXgetsense.

Value

Specified constraint senses if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**getSiftItCntCPLEX**  
*Access Total Number of Sifting Iterations*

---

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxgetsiftitcnt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
getsiftitcntcplexHenvL lpI
```

**Arguments**

- `env`  
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- `lp`  
  An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function `getSiftItCnt` which calls the CPLEX function `CPXgetsiftitcnt`.

**Value**

Zero if no solution exists, otherwise nonzero the total iteration count.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

**getSiftPase1CntCPLEX**  
*Access Number of Phase I Sifting Iterations*

---

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetsiftphase1cnt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
getSiftPase1CntCPLEX(env, lp)
```

**Arguments**

- `env`: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function `getSiftPase1Cnt` which calls the CPLEX function `CPXgetsiftphase1cnt`.

**Value**

Zero if no solution exists, otherwise nonzero the Phase I iteration count.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**

getSlackCPLEX

Accesses Slack Values for a Range of Linear Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetslack. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getSlackCPLEX(env, lp, begin, end)

Arguments

env | An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp | An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin | An integer specifying the beginning of the range of slack values to be returned.

derend | An integer specifying the end of the range of slack values to be returned.

Details

Interface to the C function getSlack which calls the CPLEX function CPXgetslack.

Value

Specified slack or surplus variables if successful, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getStatCPLEX  

Access the Solution Status of the Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetstat. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getStatCPLEX(env, lp)

Arguments

env  
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getStat which calls the CPLEX function CPXgetstat.

Value

A single integer value giving the solution status.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants section “Values returned for stat by solution”.

getStatStrCplex

*Return an Status Message String Corresponding to an Status Code*

**Description**

Low level interface function to the IBM ILOG CPLEX function `cpxgetstatstring`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```
getStatStrCplex(env, stat)
```

**Arguments**

- `env`: An object of class "*cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `stat`: The status code to be translated.

**Details**

Interface to the C function `getStatStr` which calls the CPLEX function `CPXgetstatstring`.

**Value**

A single character value containing the status message string.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- `getErrorStrCplex`
getStrParmCplex

Obtain the Current Value of a CPLEX String Parameter

Description

Low level interface function to the IBM ILOG CPLEX function cpxgetstrparm. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getStrParmCplex(env, parm)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
parm  Constant or reference number of the desired parameter.

Details

Interface to the C function getStrParm which calls the CPLEX function CPXgetstrparm.

Value

A single character value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getSubMethodCplex

Accesses Solution Method of the Last Subproblem Optimization

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsubmethod. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getSubMethodCplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSubMethod which calls the CPLEX function CPXgetsubmethod.

Value

Integer value specifying the solution method.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
getSubStatCPLex

Access Solution Status of the Last Subproblem Optimization

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetsubstat. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getSubStatCPLex(env, lp)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCPLex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCPLex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function getSubStat which calls the CPLEX function CPXgetsubstat.

Value

Zero if no solution exists, nonzero otherwise.

Author(s)

Gabriel Gelius-Dietrich <geludie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

Low level interface function to the IBM ILOG CPLEX function CPXfclose. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getTimeCLEX(env)

Arguments

env An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getTime which calls the CPLEX function CPXgettime.

Value

If successful a single numeric value, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

fileputcPLEX, openFileCLEX
getUppBndsIdsCPLEX  
*Retrieve Upper Bounds on Variables*

**Description**

The function retrieves the upper bounds on specified variables.

**Usage**

```r
getUppBndsIdsCPLEX(env, lp, ind)
```

**Arguments**

- `env`  
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

- `lp`  
  An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.

- `ind`  
  Column indices of variables (remember: first index is 0).

**Value**

A numeric vector containing the upper bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

`getUpperBndsCPLEX`
getUpperBndsCplex

Access a Range of Upper Bounds on Variables

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetub. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getUpperBndsCplex(env, lp, begin, end)

Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

begin
Beginning of the range of upper bounds to be returned.

dend
End of the range of upper bounds to be returned.

Details

Interface to the C function getUpperBnds which calls the CPLEX function CPXgetub.

Value

A numeric vector containing the lower bounds on the specified variables. If not successful an instance of class "cplexError" is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
getVersionCplex  

Get Version Number of the CPLEX Library.

Description

Low level interface function to the IBM ILOG CPLEX function CPXversion. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

getVersionCplex(env)

Arguments

env

An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

Details

Interface to the C function getVersion which calls the CPLEX function getVersionCplex.

Value

Single character string specifying the version of the cplex library or NULL if the environment does not exist.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

hybbaroptCplex  

Solve the Specified Problem by the CPLEX Barrier Optimizer

Description

Low level interface function to the IBM ILOG CPLEX function CPXhybbaropt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

hybbaroptCplex(env, lp, method)
Arguments

env
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

method
A single integer value giving the crossover method to be implemented.

Details

Interface to the C function hybbaropt which calls the CPLEX function CPXhybbaropt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

solnInfoCplex, getStatCplex, solutionCplex, cplexConstants section “LP/QP solution algorithms”.

Usage

hybnetoptCplex(env, lp, method)
Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp   An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

method A single integer value giving the type of simplex method to follow the network optimization.

Details

Interface to the C function hybnetopt which calls the CPLEX function CPXhybnetopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

solnInfoCPLEX, getStatCPLEX, solutionCPLEX, cplexConstants section “LP/QP solution algorithms”.

initProbCPLEX Create a CPLEX Problem Object in the CPLEX Environment

Description

Low level interface function to the IBM ILOG CPLEX function CPXcreateprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

initProbCPLEX(env, pname = "CPLEX_PROB", ptrtype = "cplex_prob")

Arguments

env   An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

pname A single character string containing the name of the problem object.

ptrtype A name for the pointer object.
Details

Interface to the C function CPXcreate which calls the CPLEX function CPXcreateprob.

Value

If successful, a pointer to the CPLEX problem object is returned (an instance of class "cplexPtr"), otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

delProbCplex

```r
lpoptCplex <- function(env, lp)
```

Description

Low level interface function to the IBM ILOG CPLEX function CPXlpopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

`lpoptCplex(env, lp)`

Arguments

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function lpopt which calls the CPLEX function CPXlpopt.

Value

Zero if successful, otherwise nonzero.
mipoptCPLEX

Find a Solution to a Mixed Integer Program

Description
Low level interface function to the IBM ILOG CPLEX function CPXmipopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
mipoptCLEX(env, lp)

Arguments
env An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details
Interface to the C function mipopt which calls the CPLEX function CPXmipopt.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
newColsCPLEX

Add Empty Columns to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXnewcols. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

newColsCPLEX(env, lp, ncols, 
               obj = NULL, lb = NULL, ub = NULL, 
               xctype = NULL, cnames = NULL)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **ncols**: Number of variables to add.
- **obj**: Objective function coefficients.
- **lb**: Lower bounds on the new variables.
- **ub**: Upper bounds on the new variables.
- **xctype**: Type of the new variables.
- **cnames**: Names of the new variables.

Details

Interface to the C function newCols which calls the CPLEX function CPXnewcols.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
newRowsCPLEX

Add Empty Constraints to a Specified CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXnewrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

newRowsCPLEX(env, lp, 
nrows, rhs = NULL, sense = NULL, 
rngval = NULL, rnames = NULL)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
nrows Number of new rows.
rhs Right hand side term for each new constraint.
sense Sense of each new constraint (see IBM ILOG CPLEX documentation for possible values).
rngval Range values for each new constraint.
rnames Names for the new rows.

Details

Interface to the C function newRows which calls the CPLEX function CPXnewrows.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

---

**objSaCplex**  
*Access Upper and Lower Sensitivity Ranges for Objective Function Coefficients*

### Description

Low level interface function to the IBM ILOG CPLEX function `CPXobjsA`. Consult the IBM ILOG CPLEX documentation for more detailed information.

### Usage

```r
objSaCplex(env, lp, begin, end)
```

### Arguments

- `env`: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `begin`: Beginning of the range of ranges to be returned.
- `end`: End of the range of ranges to be returned.

### Details

Interface to the C function `objSa` which calls the CPLEX function `CPXobjsA`.

### Value

If successful a list is returned:

- `lower`: lower range values
- `upper`: upper range values

otherwise an instance of class "cplexError".

### Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

### References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
openEnvCPLEX  

Initialize a CPLEX Environment

Description

Low level interface function to the IBM ILOG CPLEX function CPXopenCPLEX. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

openEnvCPLEX(ptrtype = "cplex_env")

Arguments

ptrtype  
A name for the pointer object.

Details

Interface to the C function openEnv which calls the CPLEX function CPXopenCPLEX.

Value

If successful, a pointer to the CPLEX environment is returned (an instance of class "cplexPtr"), otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldor.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

closeEnvCPLEX
openFileCPLEX  Open a File

Description

Low level interface function to the IBM ILOG CPLEX function CPXfopen. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXfopen has been removed.

Usage

openFileCPLEX(fname, ftype = "w", ptrtype = "cplex_file")

Arguments

fname  Character string giving the file name to be opened.
ftype  Character string according to the syntax of the standard C function fopen.
ptrtype  A name for the pointer object.

Details

Interface to the C function cplexfopen which calls the CPLEX function CPXfopen.

Value

A pointer to the log file (an instance of class "cplexPtr") or NULL.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

closeFileCPLEX, fileputCPLEX
Description

The function `openProbCplex` creates a new CPLEX environment and a new CPLEX problem object.

Usage

```r
openProbCplex(pname = "CPLEX_PROB",
              ptrtypeENV = "cplex_env",
              ptrtypePROB = "cplex_prob")
```

Arguments

- `pname`: A single character string containing the name of the problem object.
- `ptrtypeENV`: A name for the IBM ILOG CPLEX environment pointer object.
- `ptrtypePROB`: A name for the IBM ILOG CPLEX problem pointer object.

Details

Interface to the C functions `openEnv` and `initProb` calling CPLEX functions `cpxopenCplex` and `cpxcreateprob`.

Value

- `env`: A pointer to the CPLEX environment as returned by `openEnvCplex`.
- `lp`: A pointer to the CPLEX problem object as returned by `initProbCplex`.

If `openEnvCplex()` fails, `env` will be of class "cplexError" and `lp` will be NULL. Each list element is an object of class "cplexPtr".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References


See Also

`closeProbCplex`, `openEnvCplex`, `initProbCplex`
ordWriteCPLEX

Write Priority Order to ORD File

Description

Low level interface function to the IBM ILOG CPLEX function CPXordwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

ordWriteCPLEX(env, lp, fname)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
fname Filename.

Details

Interface to the C function ordWrite which calls the CPLEX function CPXordwrite.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldor.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
preslvWriteCplex  Write a Presolved Version of the Problem to File

Description

Low level interface function to the IBM ILOG CPLEX function CPXpreslvwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

preslvWriteCplex(env, lp, fname)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp   An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
fname Single character value giving the file name to write to.

Details

Interface to the C function preslvWrite which calls the CPLEX function CPXpreslvwrite.

Value

If successful a dingle numeric value containing the objective value difference between the original problem and the presolved problem, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgcenter/SSSA5P.

See Also

readCopyProbCplex
**presolveCPLEX**

**Perform Presolve**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXpresolve. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
presolveCPLEX(env, lp, method)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `method` A single integer value specifying the optimization algorithm to be used to solve the problem after the presolve is completed.

**Details**

Interface to the C function `presolve` which calls the CPLEX function CPXpresolve.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

cplexConstants section “LP/QP solution algorithms”.
**primoptCplex**

*Find a Solution to a Problem Using the Primal Simplex Method*

---

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXprimopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
cplexprimopt(env, lp)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp` An object of class "cplexPtr" as returned by `initProbCplex`. This is basically a pointer to an IBM ILOG CPLEX problem object.

**Details**

Interface to the C function `primopt` which calls the CPLEX function CPXprimopt.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- `solnInfoCplex`
- `getStatCplex`
- `solutionCplex`
printTerminatCPLEX  

*Print Termination Signal*

**Description**

The function `chgTerminateCPLEX` prints termination signal.

**Usage**

```c
printterminatecplex(env)
```

**Arguments**

- `env`  
  An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.

**Value**

`NULL`

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>  
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- `setterminateCPLEX`, `delterminateCPLEX`, `chgTerminateCPLEX`

---

qpoptCPLEX  

*Find a Solution to a Continuous Quadratic Program*

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXqpopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
qpoptCPLEX(env, lp)
```
Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function qpopt which calls the CPLEX function CPXqpopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

- Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgcenter/SSSA5P.

See Also

- solnInfoCplex, getStatCplex, solutionCplex

---

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXreadcopybase. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

readCopyBaseCplex(env, lp, fname)

**Arguments**

- **env**: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **fname**: Single character value giving the filename to read from.
Details

Interface to the C function readCopyBase which calls the CPLEX function CPXreadcopybase.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

readCopyMIPstartsCPLEX

Read a File in the Format MST

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopymipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

readCopyMIPstartsCPLEX(env, lp, fname)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
fname Name of the file to read from.

Details

Interface to the C function readCopyMIPstarts which calls the CPLEX function CPXreadcopymipstarts.

Value

Zero if successful, otherwise nonzero.
readCopyOrderCPLEX

**Author(s)**
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

---

**readCopyOrderCPLEX** *Read ORD File*

---

**Description**
Low level interface function to the IBM ILOG CPLEX function `cpxreadcopyorder`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**
readCopyOrderCPLEX(env, lp, fname)

**Arguments**
- env: An object of class "cplexPtr" as returned by `openEnvCPLEX`. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by `initProbCPLEX`. This is basically a pointer to an IBM ILOG CPLEX problem object.
- fname: Single character value giving the filename to read from.

**Details**
Interface to the C function `readCopyOrder` which calls the CPLEX function `CPXreadcopyorder`.

**Value**
Zero if successful, otherwise nonzero.

**Author(s)**
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
readCopyParmCPLEX

**Reads Parameter Names And Settings From a File**

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXreadcopyparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
readCopyParmCPLEX(env, fname)
```

**Arguments**

- `env` An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- `fname` Filename.

**Details**

Interface to the C function readCopyParm which calls the CPLEX function CPXreadcopyparam.

**Value**

Zero if successful, otherwise nonzero.

**Author(s)**

Gabriel Gelius-Dietrich (<geliudie@uni-duesseldorf.de>)

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- `cplexConstants`
readCopyProbCPLEX

Read an MPS, LP, or SAV File Into an Existing CPLEX Problem Object

Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopyprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

readCopyProbCPLEX(env, lp, fname, ftype = NULL)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

fname Single character value giving the filename to read from.

ftype Single character value giving the type of the file to read from.

Details

Interface to the C function readCopyProb which calls the CPLEX function CPXreadcopyprob.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

Low level interface function to the IBM ILOG CPLEX function CPXreadcopySol. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

readCopySolCplex(env, lp, fname)

Arguments

e   env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

l   lp   An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

fn   fname Single character value giving the filename to read from.

Details

Interface to the C function readCopySol which calls the CPLEX function CPXreadcopySol.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
refineConflictCPLEX  Identify a Minimal Conflict for the Infeasibility of the Linear Constraints and the Variable Bounds

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefineconflict. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

refineConflictCPLEX(env, lp)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp   An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function refineConflict which calls the CPLEX function CPXrefineconflict.

Value

If successful a list is returned:

- confnumrows  number of linear constraints in the conflict
- confnumcols  number of variable bounds in the conflict

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getConflictCPLEX
refineConflictExtCPLEX

Identify a Minimal Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefineconflictext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

refineConflictExtCPLEX(env, lp, grpcnt, concnt, 
grppref, grpbeg, grpind, grptype)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
grpcnt The number of constraint groups to be considered.
concnt Length of arrays grpind and grptype.
grppref Preferences for the groups.
grpbeg The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grpind The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grptype The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function refineConflictExt which calls the CPLEX function CPXrefineconflictext.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Refine a Conflict in Order to Determine Why a Given MIP Start is Not Feasible

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefinemipstartconflict. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

refinemipstartconflictcplex(env, lp, mipstartindex)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
mipstartindex The index of the MIP start.

Details

Interface to the C function refineMIPstartConflict which calls the CPLEX function CPXrefinemipstartconflict.

Value

If successful a list is returned:

confnumrows number of linear constraints in the conflict
confnumcols number of variable bounds in the conflict

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getcxplconflictCplex
refineMIPstartConflictExtCplex

Identify a Minimal Conflict

Description

Low level interface function to the IBM ILOG CPLEX function CPXrefinemipstartconflicttext. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

refineMIPstartConflictExtCplex(env, lp, mipstartindex, grpcnt, concnt, grppref, grpbeg, grpind, grptype)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
mipstartindex The index of the MIP start.
grpcnt The number of constraint groups to be considered.
concnt Length of arrays grpind and grptype.
grppref Preferences for the groups.
grpbeg The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grpind The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.
grptype The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.

Details

Interface to the C function refineMIPstartConflictExt which calls the CPLEX function CPXrefinemipstartconflicttext.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
return_codeCPLEX

**References**


---

return_codeCPLEX  
*Translates a IBM ILOG CPLEX Return Code into a Human Readable String*

---

**Description**

Translates a IBM ILOG CPLEX return code into a human readable string.

**Usage**

```
return_codeCPLEX(code)
```

**Arguments**

- **code**: Return (error) code from IBM ILOG CPLEX.

**Value**

An error message string corresponding to an return (error) code.

**Author(s)**

- Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**See Also**

- cplexConstants
Access Upper and Lower Sensitivity Ranges for Righthand Side Values of a Range of Constraints

Description

Low level interface function to the IBM ILOG CPLEX function CPxrhssa. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

rhsSaCplex(env, lp, begin, end)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.
- begin: Beginning of the range of ranges to be returned.
- end: End of the range of ranges to be returned.

Details

Interface to the C function rhsSa which calls the CPLEX function CPxrhssa.

Value

If successful a list is returned:

- lower: righthand side lower range values
- upper: righthand side upper range values

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
setDblParmCPLEX  

Set the Value of a CPLEX Parameter of Type Double  

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetdblparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```
setDblParmCPLEX(env, parm, value)
```

Arguments

- **env**: An object of class "cplexPtr" as returned by `openEnvCplex`. This is basically a pointer to an IBM ILOG CPLEX environment.
- **parm**: Constant or reference number of the desired parameter.
- **value**: The new value of the parameter.

Details

Interface to the C function setDblParm which calls the CPLEX function CPXsetdblparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
setDefaultParmCPLEX  
Reset All CPLEX Parameters And Settings to Default Values

Description
Low level interface function to the IBM ILOG CPLEX function CPXsetdefaults. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
setDefaultParmCPLEX(env)

Arguments
env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

Details
Interface to the C function setDefaultParm which calls the CPLEX function CPXsetdefaults.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants
Set the Value of a CPLEX Parameter of Type CPXINT

Description
Low level interface function to the IBM ILOG CPLEX function CPXsetintparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
setIntParmCplex(env, parm, value)

Arguments
env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
parm Constant or reference number of the desired parameter.
value The new value of the parameter (integer value).

Details
Interface to the C function setIntParm which calls the CPLEX function CPXsetintparam.

Value
Zero if successful, otherwise nonzero.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants
setLogFileCPLEX

Modifies the log file to which Messages are Written

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetlogfile. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality from IBM ILOG CPLEX >= 12.9.0 on, where CPXsetlogfile has been removed.

Usage

setLogFileCPLEX(env, cpfile = NULL)

Arguments

env An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.

cpfile A pointer to a file as returned by openFileCPLEX.

Details

Interface to the C function getLogFile which calls the CPLEX function CPXgetLogFile.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getLogFileCPLEX
setLogFileNameCLEX \hspace{1cm} Set and open a log file

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetlogfilename. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.8.0, where CPXsetlogfilename was not included.

Usage

```
setLogFileNameCLEX(env, filename = "cpx.log", mode = "w")
```

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **filename**: The name of the log file to open.
- **mode**: The mode in which CPLEX should open the file. The specification is the same as that for the C library function fopen. For example, use a quoted character, such as "w" to write or "a" to append. Make sure you open the file for writing; otherwise, CPLEX writes nothing to the log file, and CPLEX can produce an error every time it attempts to write. If filename is NULL, then this argument is ignored and can be NULL, too.

Details

Interface to the C function setLogFileName which calls the CPLEX function CPXsetlogfilename.

Value

Zero if successful, otherwise nonzero.

Author(s)

Mayo Roettger <mayo.roettger@hhu.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

- cplexConstants
setLongParmCplex

Set the Value of a Parameter of Type CPXLONG

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetLongParam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

setLongParmCplex(env, parm, value)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- **parm**: Constant or reference number of the desired parameter.
- **value**: New value for the parameter.

Details

Interface to the C function setLongParm which calls the CPLEX function CPXsetLongParam.

Value

Zero if successful, otherwise nonzero.

Note

In order to transfer a 64 bit integer value to CPXsetLongParam, datatype numeric is used. Parameter value is a numeric value.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

setIntParmCplex, cplexConstants
setObjDirCplex  

Change the Sense of the Optimization for a Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgobjsen. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

setObjDirCplex(env, lp, lpdire)

Arguments

env  
An object of class "cp lexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cp lexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

lpdir  
A single integer value specifying the sense of the problem.

Details

Interface to the C function setObjDir which calls the CPLEX function CPXchgobjsen.

Value

NULL

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgocenter/SSSA5P.

See Also

cplexConstants section “Generic constants”. 
setStrParmCPLEX  

Set the Value of a CPLEX String Parameter

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetstrparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

setStrParmCPLEX(env, parm, value)

Arguments

env             An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
parm            Constant or reference number of the desired parameter.
value           The new value of the parameter (character value).

Details

Interface to the C function setStrParm which calls the CPLEX function CPXsetstrparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
setTerminateCplex

Release Termination Signal

Description

Low level interface function to the IBM ILOG CPLEX function CPXsetterminate with argument terminate_p set to NULL. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

```c
setTerminateCplex(env, ptrtype = "cplex_term")
```

Arguments

- `env` An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.
- `ptrtype` A name for the pointer object.

Details

Interface to the C function delTerminate which calls the CPLEX function CPXsetterminate with argument terminate_p set to NULL.

Value

If successful, a pointer to a termination signal is returned, otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

delTerminateCplex, printTerminateCplex, chgTerminateCplex
siftoptCPLEX

Solve a Reduced Model

Description

Low level interface function to the IBM ILOG CPLEX function CPXsiftopt. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

siftoptCPLEX(env, lp)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function siftopt which calls the CPLEX function CPXsiftopt.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

Low level interface function to the IBM ILOG CPLEX function cpxsolninfo. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

solninfocplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function solnInfo which calls the CPLEX function CPXsolninfo.

Value

If successful a list is returned:

method Integer value specifying the method to produce the current solution.

type Integer value specifying the type of current solution.

primal_feasible Integer value specifying if the current solution is known to be primal feasible.

dual_feasible Integer value specifying if the current solution is known to be dual feasible.

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants, solutionCplex
solutionCplex

Access Solution Values Produced by Optimization Routines

Description

Low level interface function to the IBM ILOG CPLEX function CPLEXsolution. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

solutionCplex(env, lp)

Arguments

env  
An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp  
An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function solution which calls the CPLEX function CPLEXsolution.

Value

If successful a list is returned:

lpstat  result of the optimization
objval  objective function value
x  values of the variables for the problem
pi  values of the dual variables
slack  values of the slack or surplus variables
dj  values of the reduced costs

otherwise an instance of class "cplexError".

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants, solnInfoCplex
solWriteCplex

Write a Solution File

Description

Low level interface function to the IBM ILOG CPLEX function CPXsolwrite. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

solWriteCplex(env, lp, fname)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

fname Single character value giving the filename to write to.

Details

Interface to the C function solWrite which calls the CPLEX function CPXsolwrite.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

getProbTypeCplex, cplexConstants section “Problem Types”.

status_codeCPEX  Translates an IBM ILOG CPLEX Status Value into a Human Readable String

Description
Translates a IBM ILOG CPLEX status code into a human readable string.

Usage

status_codeCPEX(env, code)

Arguments

env  An object of class "cplexPtr" as returned by openEnvCPEX. This is basically a pointer to an IBM ILOG CPLEX environment.

code  Status code from IBM ILOG CPLEX as returned by getStatCPEX.

Value
A character string corresponding to the value of an IBM ILOG CPLEX status code as returned by getStatCPEX.

Author(s)
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References
The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also
cplexConstants, getStatStrCPEX
tightenBndsCPLEX  

Change the Lower or Upper Bounds on a Set of Variables of a Problem

Description

Low level interface function to the IBM ILOG CPLEX function CPXtightenbds. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

tightenBndsCPLEX(env, lp, ncols, ind, lu, bd)

Arguments

- env: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- lp: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- ncols: Number of bounds to be changed.
- ind: Indices of bounds to be changed.
- lu: A character vector, specifying whether an entry in bd is a upper or a lower bound on variable ind[j].
- bd: Values of the lower or upper bounds of the variables present in ind.

Details

Interface to the C function tightenBnds which calls the CPLEX function CPXtightenbds.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

chgBndsCPLEX
**tuneParmCPLEX**

*Tune Parameters of the Environment For Improved Optimizer Performance*

---

**Description**

Low level interface function to the IBM ILOG CPLEX function CPXtuneparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```c
int tuneParmCPLEX(Bcplexptr env, Bcplexptr lp, int nintp, const int* intp, const int* intpv, int ndblp, const int* dblp, const int* dblpv, int nstrp, const int* strp, const int* strpv)
```

**Arguments**

- `env`: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- `lp`: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- `nintp`: Number of integer parameters to be fixed during tuning.
- `intp`: Parameter numbers of the integer parameters which remain fixed.
- `intpv`: Values for the parameters listed in `intp`.
- `ndblp`: Number of double parameters to be fixed during tuning.
- `dblp`: Parameter numbers of the double parameters which remain fixed.
- `dblpv`: Values for the parameters listed in `dblp`.
- `nstrp`: Number of string parameters to be fixed during tuning.
- `strp`: Parameter numbers of the string parameters which remain fixed.
- `strpv`: Values for the parameters listed in `strp`.

**Details**

Interface to the C function `tuneParam` which calls the CPLEX function CPXtuneparam.

**Value**

`0` if successful, otherwise an instance of class "cplexError".

**Author(s)**

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>
unscaleProbCplex

Remove Any Scaling Applied to the Resident Problem

Description

Low level interface function to the IBM ILOG CPLEX function Cplxunscaleprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

unscaleProbCplex(env, lp)

Arguments

env An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

lp An object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

Details

Interface to the C function unscaleProb which calls the CPLEX function Cplxunscaleprob.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
writeMIPstartsCplex

Write a Range of MIP Starts to a File in MST Format

Description

Low level interface function to the IBM ILOG CPLEX function CPXwritemipstarts. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

writeMIPstartsCplex(env, lp, fname, begin, end)

Arguments

denv An object of class "cplexPtr" as returned by openEnvCplex. This is basically a pointer to an IBM ILOG CPLEX environment.

dl As an object of class "cplexPtr" as returned by initProbCplex. This is basically a pointer to an IBM ILOG CPLEX problem object.

dfname Filename to write to.

dbegin An integer specifying the beginning of the range of MIP starts to be written.

derend An integer specifying the end of the range of MIP starts to be written.

Details

Interface to the C function writeMIPstarts which calls the CPLEX function CPXwritemipstarts.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliedie@uni-duesseldorf.de>

Maintainer: Mayo Roettger <majo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
writeParmCplex

Write Names and Current Settings of CPLEX Parameters to File

Description

Low level interface function to the IBM ILOG CPLEX function CPXwriteparam. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

writeParmCplex(env, fname)

Arguments

env      An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
fname    Filename.

Details

Interface to the C function writeParm which calls the CPLEX function CPXwriteparam.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.

See Also

cplexConstants
writeProbCPLEX  Write a CPLEX Problem Object to File

Description

Low level interface function to the IBM ILOG CPLEX function CPXwriteprob. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage

writeProbCPLEX(env, lp, fname, ftype = NULL)

Arguments

- **env**: An object of class "cplexPtr" as returned by openEnvCPLEX. This is basically a pointer to an IBM ILOG CPLEX environment.
- **lp**: An object of class "cplexPtr" as returned by initProbCPLEX. This is basically a pointer to an IBM ILOG CPLEX problem object.
- **fname**: Single character value giving the file name to write to.
- **ftype**: Single character value giving the type of the file to write to.

Details

Interface to the C function writeProb which calls the CPLEX function CPXwriteprob.

Value

Zero if successful, otherwise nonzero.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
Index

*Topic optimize
addColsCPLEX, 7
addFPDestCPLEX, 9
addIndConstrCPLEX, 10
addMIPstartsCPLEX, 11
addQConstrCPLEX, 12
addRowsCPLEX, 13
baroptCPLEX, 14
baseWriteCPLEX, 15
basicPresolveCPLEX, 16
boundSaCPLEX, 17
checkAddColsCPLEX, 18
checkAddRowsCPLEX, 19
checkChgCoefListCPLEX, 20
checkCopyColTypeCPLEX, 21
checkCopyLpCPLEX, 22
checkCopyLpwNamesCPLEX, 23
checkCopyQPsepCPLEX, 25
checkCopyQuadCplex, 26
checkValsCplex, 27
chgBndsCplex, 28
chgCoefCplex, 29
chgCoefListCplex, 30
chgColNameCplex, 31
chgColsBndsCplex, 32
chgColTypeCplex, 33
chgMIPstartsCplex, 34
chgNameCplex, 35
chgObjCplex, 36
chgProbNameCplex, 37
chgProbTypeCplex, 38
chgQPcoefCplex, 39
chgRhsCplex, 40
chgRngValCplex, 41
chgRowNameCplex, 42
chgSenseCplex, 43
chgTerminateCplex, 44
cleanupCoefCplex, 44
clonProbCplex, 45
closeEnvCplex, 46
closeFileCplex, 47
closeProbCplex, 48
clpWriteCplex, 49
completeLPcplex, 50
copyBaseCplex, 51
copyColTypeCplex, 52
copyLpCplex, 53
copyLpwNamesCplex, 54
copyObjNameCplex, 55
copyOrderCplex, 56
copyPartBaseCplex, 57
copyQPsepCplex, 58
copyQuadCplex, 59
copyStartCplex, 60
cplexAPI-package, 6
cplexConstants, 61
cplexError-class, 83
cplexPtr-class, 84
delColsCplex, 85
delFpDestCplex, 86
delIndConstrsCplex, 87
delMIPstartsCplex, 88
delNamesCplex, 89
delProbCplex, 90
delQConstrsCplex, 91
delRowsCplex, 92
delSetColsCplex, 93
delSetRowsCplex, 94
delTerminateCplex, 95
disconnectChannelCplex, 96
dualoptCplex, 97
dualWriteCplex, 98
feasOptCplex, 99
filePutCplex, 100
flushChannelCplex, 101
flushStdChannelsCplex, 102
freePresolveCplex, 103
getBaseCplex, 104
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Line Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>getBestObjValCPLEX</td>
<td>105</td>
</tr>
<tr>
<td>getChannelsCLEX</td>
<td>106</td>
</tr>
<tr>
<td>getChgParmCLEX</td>
<td>107</td>
</tr>
<tr>
<td>getCoefCLEX</td>
<td>108</td>
</tr>
<tr>
<td>getColIndexCLEX</td>
<td>109</td>
</tr>
<tr>
<td>getColInfeasCLEX</td>
<td>110</td>
</tr>
<tr>
<td>getColNameCLEX</td>
<td>111</td>
</tr>
<tr>
<td>getColsCLEX</td>
<td>112</td>
</tr>
<tr>
<td>getColTypeCLEX</td>
<td>113</td>
</tr>
<tr>
<td>getConFlcCLEX</td>
<td>114</td>
</tr>
<tr>
<td>getConFlcExtCLEX</td>
<td>115</td>
</tr>
<tr>
<td>getCutoffCLEX</td>
<td>116</td>
</tr>
<tr>
<td>getDb1ParmCLEX</td>
<td>117</td>
</tr>
<tr>
<td>getDb1QualCLEX</td>
<td>118</td>
</tr>
<tr>
<td>getDbCntCLEX</td>
<td>119</td>
</tr>
<tr>
<td>getDjCLEX</td>
<td>120</td>
</tr>
<tr>
<td>getErrStrCLEX</td>
<td>121</td>
</tr>
<tr>
<td>getGradCLEX</td>
<td>122</td>
</tr>
<tr>
<td>getIndConstrCLEX</td>
<td>123</td>
</tr>
<tr>
<td>getInfoDb1ParmCLEX</td>
<td>124</td>
</tr>
<tr>
<td>getInfoIntParmCLEX</td>
<td>125</td>
</tr>
<tr>
<td>getInfoLongParmCLEX</td>
<td>126</td>
</tr>
<tr>
<td>getIntStrParmCLEX</td>
<td>127</td>
</tr>
<tr>
<td>getIntParmCLEX</td>
<td>128</td>
</tr>
<tr>
<td>getIntQualCLEX</td>
<td>129</td>
</tr>
<tr>
<td>getItCntCLEX</td>
<td>130</td>
</tr>
<tr>
<td>getLogFileCLEX</td>
<td>131</td>
</tr>
<tr>
<td>getLogFileNameCLEX</td>
<td>132</td>
</tr>
<tr>
<td>getLongParmCLEX</td>
<td>133</td>
</tr>
<tr>
<td>getLowBndsIdsCLEX</td>
<td>134</td>
</tr>
<tr>
<td>getLowerBndsCLEX</td>
<td>135</td>
</tr>
<tr>
<td>getMethodCLEX</td>
<td>136</td>
</tr>
<tr>
<td>getMIPrelGapCLEX</td>
<td>137</td>
</tr>
<tr>
<td>getMIPstartIndexCLEX</td>
<td>138</td>
</tr>
<tr>
<td>getMIPstartNameCLEX</td>
<td>139</td>
</tr>
<tr>
<td>getMIPstartsCLEX</td>
<td>140</td>
</tr>
<tr>
<td>getNumColsCLEX</td>
<td>141</td>
</tr>
<tr>
<td>getNumMIPstartsCLEX</td>
<td>142</td>
</tr>
<tr>
<td>getNumNnzCPLEX</td>
<td>143</td>
</tr>
<tr>
<td>getNumQConstrsCPLEX</td>
<td>144</td>
</tr>
<tr>
<td>getNumQnzsCPLEX</td>
<td>145</td>
</tr>
<tr>
<td>getNumQuadCPLEX</td>
<td>146</td>
</tr>
<tr>
<td>getNumRowsCPLEX</td>
<td>147</td>
</tr>
<tr>
<td>getObjCLEX</td>
<td>148</td>
</tr>
<tr>
<td>getObjDirCLEX</td>
<td>149</td>
</tr>
<tr>
<td>getObjNameCLEX</td>
<td>150</td>
</tr>
<tr>
<td>getObjOffsetCLEX</td>
<td>151</td>
</tr>
<tr>
<td>getObjValCLEX</td>
<td>152</td>
</tr>
<tr>
<td>getOrderCLEX</td>
<td>153</td>
</tr>
<tr>
<td>getParmHierNameCLEX</td>
<td>154</td>
</tr>
<tr>
<td>getParmNameCLEX</td>
<td>155</td>
</tr>
<tr>
<td>getParmNumCLEX</td>
<td>156</td>
</tr>
<tr>
<td>getParmTypeCLEX</td>
<td>157</td>
</tr>
<tr>
<td>getParmValCLEX</td>
<td>158</td>
</tr>
<tr>
<td>getPhase1CntCLEX</td>
<td>158</td>
</tr>
<tr>
<td>getPicPLEX</td>
<td>159</td>
</tr>
<tr>
<td>getPreStatCLEX</td>
<td>160</td>
</tr>
<tr>
<td>getProbNameCLEX</td>
<td>161</td>
</tr>
<tr>
<td>getProbTypeCLEX</td>
<td>162</td>
</tr>
<tr>
<td>getProbVarCLEX</td>
<td>163</td>
</tr>
<tr>
<td>getQConstrCLEX</td>
<td>164</td>
</tr>
<tr>
<td>getQPcoefCLEX</td>
<td>165</td>
</tr>
<tr>
<td>getQuadCLEX</td>
<td>166</td>
</tr>
<tr>
<td>getRedLpCLEX</td>
<td>167</td>
</tr>
<tr>
<td>getRhsCLEX</td>
<td>168</td>
</tr>
<tr>
<td>getRngValCLEX</td>
<td>169</td>
</tr>
<tr>
<td>getRowIndexCLEX</td>
<td>170</td>
</tr>
<tr>
<td>getRowInfeasCLEX</td>
<td>171</td>
</tr>
<tr>
<td>getRowNameCLEX</td>
<td>172</td>
</tr>
<tr>
<td>getRowsCLEX</td>
<td>173</td>
</tr>
<tr>
<td>getSenseCLEX</td>
<td>174</td>
</tr>
<tr>
<td>getSiftItCntCLEX</td>
<td>175</td>
</tr>
<tr>
<td>getSiftPassItCntCLEX</td>
<td>176</td>
</tr>
<tr>
<td>getSlackCLEX</td>
<td>177</td>
</tr>
<tr>
<td>getStatCLEX</td>
<td>178</td>
</tr>
<tr>
<td>getStatStrCLEX</td>
<td>179</td>
</tr>
<tr>
<td>getStrParmCLEX</td>
<td>180</td>
</tr>
<tr>
<td>getSubMethodCLEX</td>
<td>181</td>
</tr>
<tr>
<td>getSubStatCLEX</td>
<td>182</td>
</tr>
<tr>
<td>getTimeCLEX</td>
<td>183</td>
</tr>
<tr>
<td>getUppBndsIdsCLEX</td>
<td>184</td>
</tr>
<tr>
<td>getUpperBndsCLEX</td>
<td>185</td>
</tr>
<tr>
<td>getVersionCLEX</td>
<td>186</td>
</tr>
<tr>
<td>hybbaroptCLEX</td>
<td>186</td>
</tr>
<tr>
<td>hybnetoptCLEX</td>
<td>187</td>
</tr>
<tr>
<td>initProbCLEX</td>
<td>188</td>
</tr>
<tr>
<td>lpoptCLEX</td>
<td>189</td>
</tr>
<tr>
<td>mipoptCLEX</td>
<td>190</td>
</tr>
<tr>
<td>newColsCLEX</td>
<td>191</td>
</tr>
<tr>
<td>newRowsCLEX</td>
<td>192</td>
</tr>
<tr>
<td>objSaCLEX</td>
<td>193</td>
</tr>
<tr>
<td>openEnvCLEX</td>
<td>194</td>
</tr>
<tr>
<td>openFileCLEX</td>
<td>195</td>
</tr>
<tr>
<td>openProbCLEX</td>
<td>196</td>
</tr>
<tr>
<td>ordWriteCLEX</td>
<td>197</td>
</tr>
<tr>
<td>preslvWriteCPLEX</td>
<td>198</td>
</tr>
</tbody>
</table>
INDEX

presolveCPLEX, 199
primoptCPLEX, 200
printTerminateCPLEX, 201
qoptCPLEX, 201
readCopyBaseCPLEX, 202
readCopyMIPstartsCPLEX, 203
readCopyOrderCPLEX, 204
readCopyParmCPLEX, 205
readCopyProbCPLEX, 206
readCopySolCPLEX, 207
refineConflictCPLEX, 208
refineConflictExtCPLEX, 209
refineMIPstartConflictCPLEX, 210
refineMIPstartConflictExtCPLEX, 211
return_codeCPLEX, 212
rhssACPLEX, 213
setDb1ParmCPLEX, 214
setDefaultParmCPLEX, 215
setIntParmCPLEX, 216
setLogFileCPLEX, 217
setLogFileNameCPLEX, 218
setLongParmCPLEX, 219
setObjDirCPLEX, 220
setStrParmCPLEX, 221
setTerminateCPLEX, 222
siftOptCPLEX, 223
solnInfoCPLEX, 224
solutionCPLEX, 225
solWriteCPLEX, 226
status_codeCPLEX, 227
tightenBndsCPLEX, 228
tuneParmCPLEX, 229
unscaleProbCPLEX, 230
writeMIPstartsCPLEX, 231
writeParmCPLEX, 232
writeProbCPLEX, 233

*Topic package

cplexAPI-package, 6

cplexConstants, 61

copyBaseCLEX, 51
copyColTypeCLEX, 22, 52
copyLPcPLEX, 14, 23, 53
copyLPwNamesCLEX, 24, 54
copyObjNameCLEX, 55
copyOrderCLEX, 56
copyPartBaseCLEX, 57
copyQPsepCLEX, 25, 58
copyQuadCLEX, 26, 59
copyStartCLEX, 60
cplexConstants (cplexConstants), 61
cplexAPI (cplexAPI-package), 6

deadCLEX, 7, 14, 19
deadFCLEX, 9, 86
deadIndConstrCLEX, 10
deadMIPstartsCLEX, 11
deadQConstrCLEX, 12
deadRowsCLEX, 9, 13, 20

baroptCLEX, 14
baseWriteCLEX, 15

basicPresolveCLEX, 16
boundSaCLEX, 17

checkAddColsCLEX, 9, 18
checkAddRowsCLEX, 14, 19
checkChgCoeffCLEX, 20
checkCopyColTypeCLEX, 21
checkCopyLPcLEX, 22
checkCopyLPwNamesCLEX, 23
checkCopyQPsepCLEX, 25
checkCopyQuadCLEX, 26
checkValsCLEX, 27
chgBndsCLEX, 28, 32, 228
chgCoeffCLEX, 29, 39
chgCoeffListCLEX, 21, 30
chgColNameCLEX, 31
chgColsBndsCLEX, 32
chgColTypeCLEX, 33
chgMIPstartsCLEX, 34
chgNameCLEX, 35
chgObjCLEX, 29, 36
chgProbNameCLEX, 37
chgProbTypeCLEX, 38, 162
chgQPcoeffCLEX, 39
chgRhsCLEX, 29, 40
chgRngValCLEX, 14, 29, 41
chgRowNameCLEX, 42
chgSenseCLEX, 43
chgTerminateCLEX, 44, 95, 201, 222
cleanupCoeffCLEX, 44
cCloneProbCLEX, 45
closeEnvCLEX, 46, 194
closeFileCLEX, 47, 100, 195
closeProbCLEX, 48, 196
cLpWriteCLEX, 49
completeslpCLEX, 50
cplexConstantsCLEX (cplexConstants), 61
copyBaseCLEX, 51
copyColTypeCLEX, 22, 52
copyLPcPLEX, 14, 23, 53
copyLPwNamesCLEX, 24, 54
copyObjNameCLEX, 55
copyOrderCLEX, 56
copyPartBaseCLEX, 57
copyQPsepCLEX, 25, 58
copyQuadCLEX, 26, 59
copyStartCLEX, 60
cplexConstants (cplexConstants), 61
cplexAPI (cplexAPI-package), 6
cplexAPI-package, 6


cplexError (cplexError-class), 83

cplexError-class, 83

cplexPointer (cplexPtr-class), 84

cplexPointer, cplexPtr-method (cplexPtr-class), 84


cplexPtr (cplexPtr-class), 84

cplexPtr-class, 84

cplexPtrType (cplexPtr-class), 84

cplexPtrType, cplexPtr-method (cplexPtr-class), 84

cplexPtrType< (cplexPtr-class), 84

cplexPtrType<-, cplexPtr-method (cplexPtr-class), 84

CPX_ALG_AUTOMATIC (cplexConstants), 61
CPX_ALG_BAROPT (cplexConstants), 61
CPX_ALG_BARRIER (cplexConstants), 61
CPX_ALG_CONCURRENT (cplexConstants), 61
CPX_ALG_DUAL (cplexConstants), 61
CPX_ALG_FEASOPT (cplexConstants), 61
CPX_ALG_MIP (cplexConstants), 61
CPX_ALG_NET (cplexConstants), 61
CPX_ALG_NONE (cplexConstants), 61
CPX_ALG_PIVOT (cplexConstants), 61
CPX_ALG_PIVOTIN (cplexConstants), 61
CPX_ALG_PIVOTOUT (cplexConstants), 61
CPX_ALG_PRIMAL (cplexConstants), 61
CPX_ALG_ROBUST (cplexConstants), 61
CPX_ALG_SIFTING (cplexConstants), 61
CPX_AT_LOWER (cplexConstants), 61
CPX_AT_UPPER (cplexConstants), 61
CPX_BARORDER_AMD (cplexConstants), 61
CPX_BARORDER_AMF (cplexConstants), 61
CPX_BARORDER_AUTO (cplexConstants), 61
CPX_BARORDER_ND (cplexConstants), 61

CPX_BASIC (cplexConstants), 61
CPX_BASIC_SOLN (cplexConstants), 61
CPX_BINARY (cplexConstants), 61
CPX_BRANCH_DOWN (cplexConstants), 61
CPX_BRANCH_GLOBAL (cplexConstants), 61
CPX_BRANCH_UP (cplexConstants), 61
CPX_BRDIR_AUTO (cplexConstants), 61
CPX_BRDIR_DOWN (cplexConstants), 61
CPX_BRDIR_UP (cplexConstants), 61

CPX_CALLBACKCONTEXT_CANDIDATE (cplexConstants), 61

CPX_CALLBACKCONTEXT_GLOBAL_PROGRESS (cplexConstants), 61
CPX_CALLBACKCONTEXT_LOCAL_PROGRESS (cplexConstants), 61

CPX_CALLBACKCONTEXT_RELAXATION (cplexConstants), 61

CPX_CALLBACKCONTEXT_THREAD_DOWN (cplexConstants), 61
CPX_CALLBACKCONTEXT_THREAD_UP (cplexConstants), 61

CPX_CON_ABS (cplexConstants), 61
CPX_CON_DISJCT (cplexConstants), 61
CPX_CON_INDDISJCT (cplexConstants), 61
CPX_CON_INDICATOR (cplexConstants), 61
CPX_CON_LAST_CONTYPE (cplexConstants), 61

CPX_CON_LINEAR (cplexConstants), 61
CPX_CON_LOWER_BOUND (cplexConstants), 61
CPX_CON_MAXEXPR (cplexConstants), 61
CPX_CON_MINEXPR (cplexConstants), 61
CPX_CON_PWL (cplexConstants), 61
CPX_CON_QUADRATIC (cplexConstants), 61
CPX_CON_SETVAR (cplexConstants), 61
CPX_CON_SETVARCARD (cplexConstants), 61
CPX_CON_SETVARDOMAIN (cplexConstants), 61

CPX_CON_SETVAREQ (cplexConstants), 61
CPX_CON_SETVARINTERSEC (cplexConstants), 61

CPX_CON_SETVARINTERSECTION (cplexConstants), 61

CPX_CON_SETVARMAX (cplexConstants), 61
CPX_CON_SETVARMEMBER (cplexConstants), 61

CPX_CON_SETVARMIN (cplexConstants), 61
CPX_CON_SETVARNEQ (cplexConstants), 61
CPX_CON_SETVARNEQST (cplexConstants), 61
CPX_CON_SETVARNULLINTERSECT (cplexConstants, 61)
CPX_CON_SETVARSUBSET (cplexConstants, 61)
CPX_CON_SETVARSUM (cplexConstants, 61)
CPX_CON_SETVARUNION (cplexConstants, 61)
CPX_CON_SOS (cplexConstants, 61)
CPX_CON_UPPER_BOUND (cplexConstants, 61)
CPX_CONFLICT_EXCLUDED (cplexConstants, 61)
CPX_CONFLICT_LB (cplexConstants, 61)
CPX_CONFLICT_MEMBER (cplexConstants, 61)
CPX_CONFLICT_POSSIBLE_LB (cplexConstants, 61)
CPX_CONFLICT_POSSIBLE_MEMBER (cplexConstants, 61)
CPX_CONFLICT_POSSIBLE UB (cplexConstants, 61)
CPX_DPRIIND_AUTO (cplexConstants, 61)
CPX_DPRIIND_DEVEX (cplexConstants, 61)
CPX_DPRIIND_FULL (cplexConstants, 61)
CPX_DPRIIND_FULL_STEEP (cplexConstants, 61)
CPX_DPRIIND_STEEP (cplexConstants, 61)
CPX_DPRIIND_STEEPPOSTART (cplexConstants, 61)
CPX_DUAL_OBJ (cplexConstants, 61)
CPX_EXACT_KAPPA (cplexConstants, 61)
CPX_FEASOPT_MIN_INF (cplexConstants, 61)
CPX_FEASOPT_MIN_QUAD (cplexConstants, 61)
CPX_FEASOPT_MIN_SUM (cplexConstants, 61)
CPX_FEASOPT_OPT_INF (cplexConstants, 61)
CPX_FEASOPT_OPT_QUAD (cplexConstants, 61)
CPX_FEASOPT_OPT_SUM (cplexConstants, 61)
CPX_FREE_SUPER (cplexConstants, 61)
CPX_IMPLIES_INTEGER_FEASIBLE (cplexConstants, 61)
CPX_INFBOUND (cplexConstants, 61)
CPX_INTEGER (cplexConstants, 61)
CPX_INTEGER_FEASIBLE (cplexConstants, 61)
CPX_INTEGER_INFEASIBLE (cplexConstants, 61)
CPX_KAPPA (cplexConstants, 61)
CPX_KAPPA_ATTENTION (cplexConstants, 61)
CPX_KAPPA_ILLPOSED (cplexConstants, 61)
CPX_KAPPA_MAX (cplexConstants, 61)
CPX_KAPPA_STABLE (cplexConstants, 61)
CPX_KAPPA_SUSPICIOUS (cplexConstants, 61)
CPX_KAPPA_UNSTABLE (cplexConstants, 61)
CPX_MAX (cplexConstants, 61)
CPX_MAX_CMP_SLACK (cplexConstants, 61)
CPX_MAX_DUAL_INFEAS (cplexConstants, 61)
CPX_MAX_DUAL_RESIDUAL (cplexConstants, 61)
CPX_MAX_INDSLACK_INFEAS (cplexConstants, 61)
CPX_MAX_INT_INFEAS (cplexConstants, 61)
CPX_MAX_PI (cplexConstants, 61)
CPX_MAX_PRIMAL_INFEAS (cplexConstants, 61)
CPX_MAX_PRIMAL_RESIDUAL (cplexConstants, 61)
CPX_MAX_PWLSLACK_INFEAS (cplexConstants, 61)
CPX_MAX_QCPRIMAL_RESIDUAL (cplexConstants, 61)
CPX_MAX_QCSLACK (cplexConstants, 61)
CPX_MAX_QCSLACK_INFEAS (cplexConstants, 61)
CPX_MAX_RED_COST (cplexConstants, 61)
CPX_MAX_SCALED_DUAL_INFEAS (cplexConstants, 61)
CPX_MAX_SCALED_DUAL_RESIDUAL (cplexConstants, 61)
CPX_MAX_SCALED_PI (cplexConstants, 61)
CPX_MAX_SCALED_PRIMAL_INFEAS (cplexConstants, 61)
CPX_MAX_SCALED_PRIMAL_RESIDUAL (cplexConstants, 61)
CPX_MAX_SCALED_RED_COST (cplexConstants, 61)
CPX_MAX_SCALED_SLACK (cplexConstants, 61)
CPX_MAX_SCALED_X (cplexConstants, 61)
CPX_MAX_SLACK (cplexConstants, 61)
CPX_MAX_X (cplexConstants, 61)
CPX_MIN (cplexConstants, 61)
CPX_MIPEMPHASIS_BALANCED (cplexConstants, 61)
<table>
<thead>
<tr>
<th>CPX_MIPEMPHASIS_BESTBOUND</th>
<th>(cplexConstants), 61</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_MIPEMPHASIS_FEASIBILITY</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPEMPHASIS_HIDDENFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPEMPHASIS_OPTIMALITY</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPKAPPA_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPKAPPA_FULL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPKAPPA_OFF</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPKAPPA_SAMPLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPORDER_BOUNDS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPORDER_COST</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPORDER_SCALEDCOST</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSEARCH_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSEARCH_DYNAMIC</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSEARCH_TRADITIONAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSTART_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSTART_CHECKFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSTART_REPAIR</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_MIPSTART_SOLVEFIXED</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_NODEREL_BESTBOUND</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_NODEREL_BESTTEST</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_NODEREL_BESTTEST_ALT</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_NODEREL_DFS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_NONBASIC_SOLN</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_OBJ_GAP</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_OFF</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_ON</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARALLEL_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARALLEL_DETERMINISTIC</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARALLEL_OPPORTUNISTIC</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAM_ADVIND</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAM_AGGCUTLIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAM_AGGFILL</td>
<td>(cplexConstants), 61</td>
</tr>
</tbody>
</table>
CPX_PARAM_CUTSFACOR (cplexConstants), 61
CPX_PARAM_CUTUP (cplexConstants), 61
CPX_PARAM_DATACHECK (cplexConstants), 61
CPX_PARAM_DEPIND (cplexConstants), 61
CPX_PARAM_DETTLIM (cplexConstants), 61
CPX_PARAM_DISJCUTS (cplexConstants), 61
CPX_PARAM_DIVETYPE (cplexConstants), 61
CPX_PARAM_DPRIIND (cplexConstants), 61
CPX_PARAM_EACHCUTLIM (cplexConstants), 61
CPX_PARAM_EPAGAP (cplexConstants), 61
CPX_PARAM_EPGAP (cplexConstants), 61
CPX_PARAM_EPINT (cplexConstants), 61
CPX_PARAM_EPLIN (cplexConstants), 61
CPX_PARAM_EPMRK (cplexConstants), 61
CPX_PARAM_EPOPT (cplexConstants), 61
CPX_PARAM_EPOPT_H (cplexConstants), 61
CPX_PARAM_EPPER (cplexConstants), 61
CPX_PARAM_EMPRELAX (cplexConstants), 61
CPX_PARAM_EPRHS (cplexConstants), 61
CPX_PARAM_EPRHS_H (cplexConstants), 61
CPX PARAM FASTMIP (cplexConstants), 61
CPX PARAM FEASOPTMODE (cplexConstants), 61
CPX PARAM FILEENCODING (cplexConstants), 61
CPX PARAM FLOWCOVERS (cplexConstants), 61
CPX PARAM FLOWPATHS (cplexConstants), 61
CPX PARAM FPHEUR (cplexConstants), 61
CPX PARAM FRACAND (cplexConstants), 61
CPX PARAM FRACCUTS (cplexConstants), 61
CPX PARAM FRACPASS (cplexConstants), 61
CPX PARAM GUBCOVERS (cplexConstants), 61
CPX PARAM HEURFREQ (cplexConstants), 61
CPX PARAM IMPLBD (cplexConstants), 61
CPX PARAM INTSOFILEPREFIX (cplexConstants), 61
CPX PARAM INTSOFILEPREFIX (cplexConstants), 61
CPX PARAM INTSOOLLIM (cplexConstants), 61
CPX PARAM ITLIM (cplexConstants), 61
CPX PARAM LANDPCUTS (cplexConstants), 61
CPX PARAM LBHEUR (cplexConstants), 61
CPX PARAM LPMETHOD (cplexConstants), 61
CPX PARAM MCFcuts (cplexConstants), 61
CPX PARAM MEMORYEMPHASIS (cplexConstants), 61
CPX PARAM MIPCBREDLP (cplexConstants), 61
CPX PARAM MIPDISPLAY (cplexConstants), 61
CPX PARAM MIPEMPHASIS (cplexConstants), 61
CPX PARAM MIPINTERVAL (cplexConstants), 61
CPX PARAM MIPKAPPASTATS (cplexConstants), 61
CPX PARAM MIPORDIND (cplexConstants), 61
CPX PARAM MIPORDTYPE (cplexConstants), 61
CPX PARAM MIPSEARCH (cplexConstants), 61
CPX PARAM MIQPSTRAT (cplexConstants), 61
CPX PARAM MIRCUTS (cplexConstants), 61
CPX PARAM MPSLONGLIM (cplexConstants), 61
CPX PARAM NETDISPLAY (cplexConstants), 61
CPX PARAM NETEPOPT (cplexConstants), 61
CPX PARAM NETEPRHS (cplexConstants), 61
CPX PARAM NETFIND (cplexConstants), 61
CPX PARAM NETFILIM (cplexConstants), 61
CPX PARAM NETPPIIND (cplexConstants), 61
CPX PARAM NODEFILEIND (cplexConstants), 61
CPX PARAM NODELIM (cplexConstants), 61
CPX PARAM NODESEL (cplexConstants), 61
CPX PARAM NUMERICALEMPHASIS (cplexConstants), 61
CPX PARAM NZREADLIM (cplexConstants), 61
CPX PARAM OBJJDF (cplexConstants), 61
CPX PARAM OBJLIM (cplexConstants), 61
CPX PARAM OBULIM (cplexConstants), 61
CPX PARAM PARALLELMODE (cplexConstants), 61
CPX PARAM PARAMDISPLAY (cplexConstants), 61
CPX PARAM PERIND (cplexConstants), 61
CPX PARAM PERLIM (cplexConstants), 61
CPX PARAM POLISHAFTERDETTIME (cplexConstants), 61
CPX PARAM POLISHAFTERPAG (cplexConstants), 61
CPX PARAM POLISHAFTEREPAG (cplexConstants), 61
INDEX

CPX_PARAM_POLISHAFTERINTSOL (cplexConstants), 61
CPX_PARAM_POLISHAFTERNODE (cplexConstants), 61
CPX_PARAM_POLISHAFTERTIME (cplexConstants), 61
CPX_PARAM_POLISHTIME (cplexConstants), 61
CPX_PARAM_POPULATELIM (cplexConstants), 61
CPX_PARAM_PPRIIND (cplexConstants), 61
CPX_PARAM_PREDUAL (cplexConstants), 61
CPX_PARAM_PREIND (cplexConstants), 61
CPX_PARAM_PRELINEAR (cplexConstants), 61
CPX_PARAM_PREPASS (cplexConstants), 61
CPX_PARAM_PRESLVDND (cplexConstants), 61
CPX_PARAM_PRICELIM (cplexConstants), 61
CPX_PARAM_PROBE (cplexConstants), 61
CPX_PARAM_PROBEDETTIME (cplexConstants), 61
CPX_PARAM_PROBETIME (cplexConstants), 61
CPX_PARAM_QPMAKEPSDIND (cplexConstants), 61
CPX_PARAM_QPMETHOD (cplexConstants), 61
CPX_PARAM_QPNRREADLIN (cplexConstants), 61
CPX_PARAM_RAMUPDDETTILIM (cplexConstants), 61
CPX_PARAM_RAMUPDURATION (cplexConstants), 61
CPX_PARAM_RAMUPTILIM (cplexConstants), 61
CPX_PARAM_RANDOMSEED (cplexConstants), 61
CPX_PARAM_REDUCE (cplexConstants), 61
CPX_PARAM_REINV (cplexConstants), 61
CPX_PARAM_RELAXPREIND (cplexConstants), 61
CPX_PARAM_RELOBJDIF (cplexConstants), 61
CPX_PARAM_REPAIRTRIES (cplexConstants), 61
CPX_PARAM_REPEATPRESOLVE (cplexConstants), 61
CPX_PARAM.ReverseIND (cplexConstants), 61
CPX_PARAM_RFICLEMUL (cplexConstants), 61
CPX_PARAM_RINSHEUR (cplexConstants), 61
CPX_PARAM_ROWREADLIN (cplexConstants), 61
CPX_PARAM_SCAIND (cplexConstants), 61
CPX_PARAM_SCRIND (cplexConstants), 61
CPX_PARAM_SIFTAG (cplexConstants), 61
CPX_PARAM_SIFTDISPLAY (cplexConstants), 61
CPX_PARAM_SIFTTILIM (cplexConstants), 61
CPX_PARAM_SIMDISPLAY (cplexConstants), 61
CPX_PARAM_SINGLIM (cplexConstants), 61
CPX_PARAM_SINGTOL (cplexConstants), 61
CPX_PARAM_SOLNPOOLAGAP (cplexConstants), 61
CPX_PARAM_SOLNPOOLCAPACITY (cplexConstants), 61
CPX_PARAM_SOLNPOOLGAP (cplexConstants), 61
CPX_PARAM_SOLNPOOLINTENSITY (cplexConstants), 61
CPX_PARAM_SOLNPOOLREPLACE (cplexConstants), 61
CPX_PARAM_SOLUTIONTARGET (cplexConstants), 61
CPX_PARAM_STARTALG (cplexConstants), 61
CPX_PARAM_STRONGCANDLIM (cplexConstants), 61
CPX_PARAM_STRONGTILIM (cplexConstants), 61
CPX_PARAM_SUBALG (cplexConstants), 61
CPX_PARAM_SUBMINODELIM (cplexConstants), 61
CPX_PARAM_SYMOMETRY (cplexConstants), 61
CPX_PARAM_THREADS (cplexConstants), 61
CPX_PARAM_TILIM (cplexConstants), 61
CPX_PARAM_TRELIM (cplexConstants), 61
CPX_PARAM_TUNINGDETTILIM (cplexConstants), 61
CPX_PARAM_TUNINGDISPLAY (cplexConstants), 61
CPX_PARAM_TUNINGMEASURE (cplexConstants), 61
CPX_PARAM_TUNINGREPEAT (cplexConstants), 61
CPX_PARAM_TUNINGTILIM (cplexConstants), 61
CPX_PARAM_VAREL (cplexConstants), 61
CPX_PARAM_WORKDIR (cplexConstants), 61
CPX PARAM WORKMEM (cplexConstants), 61
<table>
<thead>
<tr>
<th>Index</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_PARAM_WRITELEVEL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAM_XXXIND</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAM_ZEROHALFCUTS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_DOUBLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_INTC</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_LONG</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_NONE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PARAMTYPE_STRING</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_DEVE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_FULL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_PARTIAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_STEEP</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PPRIIND_STEEPPOSTSTART</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRECOL_AGG</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRECOL_FIX</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRECOL_LOW</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRECOL_OTHER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRECOL_UP</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREREDUCE_DUALONLY</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREREDUCE_NOPRIMALORDUAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREREDUCE_PRIMALANDDUAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREREDUCE_PRIMALONLY</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREROW_AGG</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREROW_OTHER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PREROW_RED</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRIMAL_OBJ</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_PRIMAL_SOLN</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SEMICONT</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SEMIINT</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLNPPOOL_DIV</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLNPPOOL_FIFO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLNPPOOL_FILTER_DIVERSITY</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLNPPOOL_FILTER_RANGE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLNPPOOL_OBJ</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_FIRSTORDER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_OPTIMALCONVEX</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_OPTIMALGLOBAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_DETIME_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_DUAL_OBJ_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_IT_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_OBJ_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_PRIM_OBJ_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_TIME_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_ABORT_USER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_CONTRADITION</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_DETIME_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_IT_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_MEM_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_NODE_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_OBJ_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_TIME_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_ABORT_USER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_FEASIBLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_CONFLICT_MINIMAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_INF</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_QUAD</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_FEASIBLE_RELAXED_SUM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_FIRSTORDER</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPX_STAT_INFEASIBLE</td>
<td>(cplexConstants), 61</td>
</tr>
</tbody>
</table>
INDEX

CPXcheckcputype
   (checkCopyColTypeCplex), 21
CPXcheckcopylp (checkCopyLpCplex), 22
CPXcheckcopylpnames
   (checkCopyLpwNamesCplex), 23
CPXcheckcopypssep
   (checkCopyQPsepCplex), 25
CPXcheckcopyquad (checkCopyQuadCplex), 26
CPXcheckvals (checkValsCplex), 27
CPXchgbds (chgBndsCplex), 28
CPXchgcoef (chgCoeffCplex), 29
CPXchgcoeflist (chgCoeffListCplex), 30
CPXchgcolname (chgColNameCplex), 31
CPXchgctype (chgColTypeCplex), 33
CPXchgMipstarts (chgMipstartsCplex), 34
CPXchgname (chgNameCplex), 35
CPXchgbobj (chgObjCplex), 36
CPXchgbobjjsen (setObjDirCplex), 220
CPXchgprobname (chgProbNameCplex), 37
CPXchgprobtype (chgProbTypeCplex), 38
CPXchgqcobj (chgQcobjCplex), 39
CPXchgrhs (chgRhsCplex), 40
CPXchgrngval (chgRngValCplex), 41
CPXchgrwncolnames (chgRowNamesCplex), 42
CPXchgsense (chgSenseCplex), 43
CPXcleancleanup (cleanupCoeffCplex), 44
CPXcloneprob (cloneProbCplex), 45
CPXcloseCplex (closeEnvCplex), 46
CPXclpwrite (clpWriteCplex), 49
CPXcompletelp (completelpCplex), 50
CPXcopybase (copyBaseCplex), 51
CPXcopyctype (copyColTypeCplex), 52
CPXcopylp (copyLpCplex), 53
CPXcopylpnames (copyLpwNamesCplex), 54
CPXcopyobjname (copyObjNameCplex), 55
CPXcopyorder (copyOrderCplex), 56
CPXcopypartialbase (copyPartBaseCplex), 57
CPXcopyqseep (copyQPsepCplex), 58
CPXcopyquad (copyQuadCplex), 59
CPXcopystart (copyStartCplex), 60
CPXcreateprob (initProbCplex), 188
CPXdelcols (delColsCplex), 85
CPXdelfpdest (delFpDestCplex), 86
CPXdelindconstrs (delIndConstrsCplex), 87
CPXdelmipstarts (delMipstartsCplex), 88
CPXdelnames (delNamesCplex), 89
CPXdelqconstrs (delQConstrsCplex), 91
CPXdelrows (delRowsCplex), 92
CPXdelsetcols (delSetColsCplex), 93
CPXdelsetrows (delSetRowsCplex), 94
CPXdisconnectchannel
   (disconnectChannelCplex), 96
CPXdualopt (dualoptCplex), 97
CPXdualwrite (dualWriteCplex), 98
CPXERR_BAD_MULTIOBJ_ATTR
   (cplexConstants), 61
CPXERR_CALLBACK_INCONSISTENT
   (cplexConstants), 61
CPXERR_CAND_NOT_POINT (cplexConstants), 61
CPXERR_CAND_NOTRAY (cplexConstants), 61
CPXERR_MULTIOBJ_SUBPROB_SOLVE
   (cplexConstants), 61
CPXERR_NEGATIVE_SURPLUS
   (cplexConstants), 61
CPXERR_NO_OBJ_NAME (cplexConstants), 61
CPXERR_NO_SENSIT (cplexConstants), 61
CPXERR_NOT_FOR_MULTIOBJ
   (cplexConstants), 61
CPXfclose (closeFileCplex), 47
CPXfeasopt (feasOptCplex), 99
CPXflushchannel (flushChannelCplex), 101
CPXflushstdchannels
   (FlushStdChannelsCplex), 102
CPXfopen (openFileCplex), 195
CPXfpults (fileputCplex), 100
CPXfreeresolve (freeResolveCplex), 103
CPXfreeprob (delProbCplex), 90
CPXgetbase (getBaseCplex), 104
CPXgetbestobjval (getBestObjValCplex), 105
CPXgetchannels (getChannelsCplex), 106
CPXgetchparam (getChgParmCplex), 107
CPXgetcoef (getCoeffCplex), 108
CPXgetcolindex (getColIndexCplex), 109
CPXgetcolfeas (getColFeasCplex), 110
CPXgetcolname (getColNameCplex), 111
CPXgetcols (getColsCplex), 112
CPXgetconflict (getConflictCplex), 114
CPXgetconflictext
   (getConflictTextCplex), 115
CPXgetctype (getColTypeCplex), 113
CPXgetcutoff (getCutoffCplex), 116
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPXMIP_ABORT_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_ABORT_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_ABORT_RELAXED</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_DETTIME_LIM_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_DETTIME_LIM_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FAIL_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FAIL_FEAS_NO_TREE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FAIL_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FAIL_INFEAS_NO_TREE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FEASIBLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FEASIBLE_RELAXED_INF</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FEASIBLE_RELAXED_QUAD</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_FEASIBLE_RELAXED_SUM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_INFEASIBLE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_INFORUNBD</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_MEM_LIM_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_MEM_LIM_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_NODE_LIM_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_NODE_LIM_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_POPULATED</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_POPULATED_TOL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_RELAXED_INF</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_RELAXED_QUAD</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_RELAXED_SUM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_OPTIMAL_TOL</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_POPULATESOL_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_SOL_LIM</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_TIME_LIM_FEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_TIME_LIM_INFEAS</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXMIP_UNBOUNDED</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXmipopt</td>
<td>(mipoptCPLEX), 190</td>
</tr>
<tr>
<td>CPXNET_NO_DISPLAY_OBJECTIVE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXNET_PENALIZED_OBJECTIVE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXNET_PRICE_AUTO</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXNET_PRICE_MULTI_PART</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXNET_PRICE_SORT_MULTI_PART</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXNET_TRUE_OBJECTIVE</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXnewcols</td>
<td>(newColsCplex), 191</td>
</tr>
<tr>
<td>CPXnewrows</td>
<td>(newRowsCplex), 192</td>
</tr>
<tr>
<td>CPXobjsa</td>
<td>(objSaCplex), 193</td>
</tr>
<tr>
<td>CPXopenCplex</td>
<td>(openEnvCplex), 194</td>
</tr>
<tr>
<td>CPXordwrite</td>
<td>(ordWriteCplex), 197</td>
</tr>
<tr>
<td>CPXPARAM_Advance</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Algorithm</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_ColNonzeros</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_ConvergeTol</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Crossover</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Display</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Limits_Corrections</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Limits_Growth</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Limits_Iteration</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Limits_ObjRange</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_Ordering</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_QCPConvergeTol</td>
<td>(cplexConstants), 61</td>
</tr>
<tr>
<td>CPXPARAM_Barrier_StartAlg</td>
<td>(cplexConstants), 61</td>
</tr>
</tbody>
</table>
CPXPARAM_ClockType (cplexConstants), 61
CPXPARAM_Conflict_Display (cplexConstants), 61
CPXPARAM_DetTimeLimit (cplexConstants), 61
CPXPARAM_DistMIP_Rampup_DetTimeLimit (cplexConstants), 61
CPXPARAM_DistMIP_Rampup_Duration (cplexConstants), 61
CPXPARAM_DistMIP_Rampup_TimeLimit (cplexConstants), 61
CPXPARAM_Empasis_Memory (cplexConstants), 61
CPXPARAM_Empasis_MIP (cplexConstants), 61
CPXPARAM_Empasis_Numerical (cplexConstants), 61
CPXPARAM_Feasopt_Mode (cplexConstants), 61
CPXPARAM_Feasopt_Tolerance (cplexConstants), 61
CPXPARAM_LPMethod (cplexConstants), 61
CPXPARAM_MIP_Cuts_Cliques (cplexConstants), 61
CPXPARAM_MIP_Cuts_Covers (cplexConstants), 61
CPXPARAM_MIP_Cuts_Disjunctive (cplexConstants), 61
CPXPARAM_MIP_Cuts_FlowCovers (cplexConstants), 61
CPXPARAM_MIP_Cuts_Gomory (cplexConstants), 61
CPXPARAM_MIP_Cuts_GUBCovers (cplexConstants), 61
CPXPARAM_MIP_Cuts_Implied (cplexConstants), 61
CPXPARAM_MIP_Cuts_LiftProj (cplexConstants), 61
CPXPARAM_MIP_Cuts_MCFCut (cplexConstants), 61
CPXPARAM_MIP_Cuts_MICCut (cplexConstants), 61
CPXPARAM_MIP_Cuts_PathCut (cplexConstants), 61
CPXPARAM_MIP_Cuts_ZeroHalfCut (cplexConstants), 61
CPXPARAM_MIP_Display (cplexConstants), 61
CPXPARAM_MIP_Interval (cplexConstants), 61
CPXPARAM_MIP_Limits_AggForCut (cplexConstants), 61
CPXPARAM_MIP_Limits_AuxRootThreads (cplexConstants), 61
CPXPARAM_MIP_Limits_CutPasses (cplexConstants), 61
CPXPARAM_MIP_Limits_CutsFactor (cplexConstants), 61
CPXPARAM_MIP_Limits_EachCutLimit (cplexConstants), 61
CPXPARAM_MIP_Limits_GomoryCand (cplexConstants), 61
CPXPARAM_MIP_Limits_GomoryPass (cplexConstants), 61
CPXPARAM_MIP_Limits_Nodes (cplexConstants), 61
CPXPARAM_MIP_Limits_PolishTime (cplexConstants), 61
CPXPARAM_MIP_Limits_Populate (cplexConstants), 61
CPXPARAM_MIP_Limits_ProbeDetTime (cplexConstants), 61
CPXPARAM_MIP_Limits_ProbeTime (cplexConstants), 61
CPXPARAM_MIP_Limits_RepairTries (cplexConstants), 61
CPXPARAM_MIP_Limits_Solutions (cplexConstants), 61
CPXPARAM_MIP_Limits_StrongCand (cplexConstants), 61
CPXPARAM_MIP_Limits_StrongIt (cplexConstants), 61
CPXPARAM_MIP_Limits_SubMIPNodeLim (cplexConstants), 61
CPXPARAM_MIP_Limits_TreeMemory (cplexConstants), 61
CPXPARAM_MIP_OrderType (cplexConstants), 61
CPXPARAM_MIP_POLISHAFTER_AbsMIPGap (cplexConstants), 61
CPXPARAM_MIP_POLISHAFTER_DepTime (cplexConstants), 61
CPXPARAM_MIP_POLISHAFTER_MIPGap (cplexConstants), 61
CPXPARAM_MIP_POLISHAFTER_Nodes (cplexConstants), 61
INDEX

CPXPARAM_MIP_PolishAfter_Solutions
  (cplexConstants), 61
CPXPARAM_MIP_PolishAfter_Time
  (cplexConstants), 61
CPXPARAM_MIP_Pool_AbsGap
  (cplexConstants), 61
CPXPARAM_MIP_Pool_Capacity
  (cplexConstants), 61
CPXPARAM_MIP_Pool_Intensity
  (cplexConstants), 61
CPXPARAM_MIP_Pool_RelGap
  (cplexConstants), 61
CPXPARAM_MIP_Pool_Replace
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Backtrack
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_BBInterval
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Branch
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_CallbackReducedLP
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Dive
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_File
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_FPHeur
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_HeuristicFreq
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_KappaStats
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_LBHeur
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_MIQCPStrat
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_NodeSelect
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Order
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_PresolveNode
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Probe
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_RINSHeur
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_Search
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_StartAlgorithm
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_SubAlgorithm
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_SubMIPScale
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_SubMIPStartAlg
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_SubMIPSubAlg
  (cplexConstants), 61
CPXPARAM_MIP_Strategy_VariableSelect
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_AbsMIPGap
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_Integality
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_LowerCutoff
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_MIPGap
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_ObjDifference
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_RelObjDifference
  (cplexConstants), 61
CPXPARAM_MIP_Tolerances_UpperCutoff
  (cplexConstants), 61
CPXPARAM_MultiObjective_Display
  (cplexConstants), 61
CPXPARAM_Network_Display
  (cplexConstants), 61
CPXPARAM_Network_Iterations
  (cplexConstants), 61
CPXPARAM_Network_NetFind
  (cplexConstants), 61
CPXPARAM_Network_Pricing
  (cplexConstants), 61
CPXPARAM_Network_Tolerances_Feasibility
  (cplexConstants), 61
CPXPARAM_Network_Tolerances_Optimality
  (cplexConstants), 61
CPXPARAM_Output_CloneLog
  (cplexConstants), 61
CPXPARAM_Output_IntSolFilePrefix
  (cplexConstants), 61
CPXPARAM_Output_MPSLong
  (cplexConstants), 61
CPXPARAM_Output_WriteLevel
  (cplexConstants), 61
INDEX

CPXPARAM_Parallel (cplexConstants), 61
CPXPARAM_ParamDisplay (cplexConstants), 61
CPXPARAM_Preprocessing_Aggregator (cplexConstants), 61
CPXPARAM_Preprocessing_BoundStrength (cplexConstants), 61
CPXPARAM_Preprocessing_CoeffReduce (cplexConstants), 61
CPXPARAM_Preprocessing_Dependency (cplexConstants), 61
CPXPARAM_Preprocessing_Dual (cplexConstants), 61
CPXPARAM_Preprocessing_Fill (cplexConstants), 61
CPXPARAM_Preprocessing_Folding (cplexConstants), 61
CPXPARAM_Preprocessing_Linear (cplexConstants), 61
CPXPARAM_Preprocessing_NumPass (cplexConstants), 61
CPXPARAM_Preprocessing_Presolve (cplexConstants), 61
CPXPARAM_Preprocessing_QCPDuals (cplexConstants), 61
CPXPARAM_Preprocessing_QPMakePSD (cplexConstants), 61
CPXPARAM_Preprocessing_HReduce (cplexConstants), 61
CPXPARAM_Preprocessing_Relax (cplexConstants), 61
CPXPARAM_Preprocessing_RepeatePresolve (cplexConstants), 61
CPXPARAM_Preprocessing_Symmetry (cplexConstants), 61
CPXPARAM_QPMethod (cplexConstants), 61
CPXPARAM_RandomSeed (cplexConstants), 61
CPXPARAM_Read_APIEncoding (cplexConstants), 61
CPXPARAM_Read_Constraints (cplexConstants), 61
CPXPARAM_Read_DataCheck (cplexConstants), 61
CPXPARAM_Read_FileEncoding (cplexConstants), 61
CPXPARAM_Read_Nonzeros (cplexConstants), 61
CPXPARAM_Read_QPNonzeros (cplexConstants), 61
CPXPARAM_Read_Scale (cplexConstants), 61
CPXPARAM_Read_Variables (cplexConstants), 61
CPXPARAM_Read_WarningLimit (cplexConstants), 61
CPXPARAM_Record (cplexConstants), 61
CPXPARAM_ScreenOutput (cplexConstants), 61
CPXPARAM_Sifting_Algorithm (cplexConstants), 61
CPXPARAM_Sifting_Display (cplexConstants), 61
CPXPARAM_Sifting_Iterations (cplexConstants), 61
CPXPARAM_Simplex_Crash (cplexConstants), 61
CPXPARAM_Simplex_DGradient (cplexConstants), 61
CPXPARAM_Simplex_Display (cplexConstants), 61
CPXPARAM_Simplex_Limits_Iterations (cplexConstants), 61
CPXPARAM_Simplex_Limits_LowerObj (cplexConstants), 61
CPXPARAM_Simplex_Limits_Perturbation (cplexConstants), 61
CPXPARAM_Simplex_Limits_Singularity (cplexConstants), 61
CPXPARAM_Simplex_Limits_UpperObj (cplexConstants), 61
CPXPARAM_Simplex_Perturbation_Constant (cplexConstants), 61
CPXPARAM_Simplex_Perturbation_Indicator (cplexConstants), 61
CPXPARAM_Simplex_PGradient (cplexConstants), 61
CPXPARAM_Simplex_Pricing (cplexConstants), 61
CPXPARAM_Simplex_Refactor (cplexConstants), 61
CPXPARAM_Simplex_Tolerances_Feasibility (cplexConstants), 61
CPXPARAM_Simplex_Tolerances_Markowitz (cplexConstants), 61
CPXPARAM_Simplex_Tolerances_Optimality (cplexConstants), 61
CPXPARAM_SolutionTarget
(cplexConstants), 61
CPXPARAM_Threads (cplexConstants), 61
CPXPARAM_TimeLimit (cplexConstants), 61
CPXPARAM_Tune_DetTimeLimit (cplexConstants), 61
CPXPARAM_Tune_Display (cplexConstants), 61
CPXPARAM_Tune_Measure (cplexConstants), 61
CPXPARAM_Tune_Repeat (cplexConstants), 61
CPXPARAM_Tune_TimeLimit (cplexConstants), 61
CPXPARAM_WorkDir (cplexConstants), 61
CPXPARAM_WorkMem (cplexConstants), 61
CPXpresolvewrite (presolveWriteCPLEX), 198
CPXprimsolve (primsolveCPLEX), 199
CPXprimopt (primoptCPLEX), 200
CPXPROB_FIXEDMILP (cplexConstants), 61
CPXPROB_FIXEDMIQP (cplexConstants), 61
CPXPROB_LP (cplexConstants), 61
CPXPROB_MILP (cplexConstants), 61
CPXPROB_MIQCP (cplexConstants), 61
CPXPROB_MIQP (cplexConstants), 61
CPXPROB_NODELP (cplexConstants), 61
CPXPROB_NODEQCP (cplexConstants), 61
CPXPROB_QCP (cplexConstants), 61
CPXPROB_OP (cplexConstants), 61
CPXqpopt (qpoptCPLEX), 201
CPXreadcopybase (readCopyBaseCPLEX), 202
CPXreadcopymipstarts (readCopyMipStartsCPLEX), 203
CPXreadcopyorder (readCopyOrderCPLEX), 204
CPXreadcopyparam (readCopyParamCPLEX), 205
CPXreadcopyprob (readCopyProbCPLEX), 206
CPXreadcopysolution (readCopySolutionCPLEX), 207
CPXrefineconflict (refineConflictCPLEX), 208
CPXrefineconflicttext (refineConflictTextCPLEX), 209
CPXrefinemipstartconflict (refineMipStartConflictCPLEX), 210
CPXrefinemipstartconflicttext (refineMipStartConflictTextCPLEX), 211
CPXrhssa (rhsSAcPLEX), 213
CPXsetdblparam (setDblParamCPLEX), 214
CPXsetdefaults (setDefaultParamCPLEX), 215
CPXsetintparam (setIntParamCPLEX), 216
CPXsetlogfile (setLogFileCPLEX), 217
CPXsetlogfile (setLogFileCPLEX), 218
CPXsetlongparam (setLongParamCPLEX), 219
CPXsetstrparam (setStrParamCPLEX), 221
CPXsetterminate (setTerminateCPLEX), 222
CPXsiftopt (siftoptCPLEX), 223
CPXsolinfo (solInfoCPLEX), 224
CPXsolution (solutionCPLEX), 225
CPXsolwrite (solWriteCPLEX), 226
CPXtightenbnds (tightenBndsCPLEX), 228
CPXtuneparam (tuneParamCPLEX), 229
CPXunscaleprob (unscaleProbCPLEX), 230
CPXversion (getVersionCPLEX), 231
CPXwritemipstarts (writeMipStartsCPLEX), 232
CPXwriteparam (writeParamCPLEX), 233
delColsCPLEX, 85
delFpDestCPLEX, 9, 86
delIndConstrsCPLEX, 87
delMipstartsCPLEX, 88
delNamesCPLEX, 89
delProbCPLEX, 90, 189
delQConstrsCPLEX, 91
delRowsCPLEX, 92
delSetColsCPLEX, 93
delSetRowsCPLEX, 94
delTerminateCPLEX, 44, 95, 201, 222
disconnectChannelCPLEX, 96, 101, 102, 106
dualOptCPLEX, 97
dualWriteCPLEX, 98
er (cplexError-class), 83
er, cplexError-method (cplexError-class), 83
ermsg (cplexError-class), 83
ermsg, cplexError-method (cplexError-class), 83
ernum (cplexError-class), 83
ernum, cplexError-method (cplexError-class), 83
errnum<-(cplexError-class), 83
errnum<-,cplexError-method
  (cplexError-class), 83

feasOptCPLEX, 99
fileputcplex, 47, 100, 183, 195
flushChannelsCPLEX, 96, 101, 102, 106
flushStdChannelsCPLEX, 96, 101, 102, 106
freePresolveCPLEX, 103
getBaseCPLEX, 104
getBestObjValCPLEX, 105, 137
getChannelsCPLEX, 96, 101, 102, 106
getChgParmCPLEX, 107, 158
getCoefCPLEX, 108
getColIndexCPLEX, 109
getColInfeasCPLEX, 99, 110
getColNameCPLEX, 111
getColsCPLEX, 112
getColTypeCPLEX, 113
getConFLICTCPLEX, 114, 208, 210
getConFLICTExtCPLEX, 115
getCutoffCPLEX, 116
getDb1ParmCPLEX, 117
genDb1QualCPLEX, 118
getDbsCntCPLEX, 119
getDjCPLEX, 120
getAddressCPLEX, 121, 179
getGradCPLEX, 122
genIndConstrCPLEX, 123
getInfoDb1ParmCPLEX, 124
getInfoIntParmCPLEX, 125, 127
getInfoLongParmCPLEX, 126
getInfoStrParmCPLEX, 127
genIntParmCPLEX, 128, 133
getIntQualCPLEX, 129
genIntCntCPLEX, 130
genLogFileCPLEX, 131, 217
genLogFileNameCPLEX, 132
genLongParmCPLEX, 133
genLowBndsIdsCPLEX, 134
genLowerBndsCPLEX, 28, 134, 135
genMethodCPLEX, 136
genMPre1GapCPLEX, 137
genMIPStartIndexCPLEX, 138
genMIPStartNameCPLEX, 139
genMIPstartsCPLEX, 140
getNumColsCPLEX, 29, 141
genNumMIPstartsCPLEX, 142
genNumNnzCPLEX, 143
getNumQConstrsCPLEX, 144
getNumQPnzCPLEX, 145
genNumQuadCPLEX, 146
genNumRowsCPLEX, 29, 147
getObjCPLEX, 148
genObjDirCPLEX, 149
genObjNameCPLEX, 150
genObjOffsetCPLEX, 151
genObjValCPLEX, 137, 152
getOrderCPLEX, 153
getParmHierNameCPLEX, 154
getParmNameCPLEX, 155
getParmNumCPLEX, 156
getParmTypeCPLEX, 157
getParmValCPLEX, 83, 158
getPhase1CntCPLEX, 158
getPiCPLEX, 159
getPreStatCPLEX, 160
genProbNameCPLEX, 161
genProbTypeCPLEX, 38, 162, 226
genProbVarCPLEX, 163
getQConstrCPLEX, 164
genQPcoefCPLEX, 165
genQuadCPLEX, 166
getRedLpCPLEX, 167
getRhsCPLEX, 168
genRowValCPLEX, 169
genRowIndexCPLEX, 170
getRowInfeasCPLEX, 99, 171
genRowNameCPLEX, 172
genRowsCPLEX, 173
genSenseCPLEX, 174
genSiftItCntCPLEX, 175
genSiftPase1CntCPLEX, 176
getSlackCPLEX, 177
getStatCPLEX, 15, 97, 99, 178, 187, 188, 190,
  191, 200, 202, 227
genStatStrCPLEX, 127, 179, 227
getStrParmCPLEX, 180
getSubMethodCPLEX, 181
getSubStatCPLEX, 182
genTimeCPLEX, 183
genUppBndsIdsCPLEX, 184
getUpperBndsCPLEX, 28, 184, 185
genVersionCPLEX, 186
hybbaroptCPLEX, 186
hybnetoptCPLEX, 187
isCpleXchanPointer (cplexPtr-class), 84
isCpleXchanPointer, cplexPtr-method (cplexPtr-class), 84
isCplexEnvPointer (cplexPtr-class), 84
isCplexEnvPointer, cplexPtr-method (cplexPtr-class), 84
isCplexFilePointer (cplexPtr-class), 84
isCplexFilePointer, cplexPtr-method (cplexPtr-class), 84
isCplexProbPointer (cplexPtr-class), 84
isCplexProbPointer, cplexPtr-method (cplexPtr-class), 84
isCplexTermPointer (cplexPtr-class), 84
isCplexTermPointer, cplexPtr-method (cplexPtr-class), 84
isNULLpointerCplex (cplexPtr-class), 84
isNULLpointerCplex, cplexPtr-method (cplexPtr-class), 84
lpoptCplex, 189
mipoptCplex, 190
newColsCplex, 191
newRowsCplex, 192
objSaCplex, 193
openFileCplex, 9, 47, 86, 100, 183, 195, 217
openProbCplex, 48, 196
ordWriteCplex, 197
preslvWriteCplex, 198
presolveCplex, 199
primoptCplex, 200
printTerminateCplex, 44, 95, 201, 222
qoptCplex, 201
readCopyBaseCplex, 202
readCopyMIPstartsCplex, 203
readCopyOrderCplex, 204
readCopyParmCplex, 205
readCopyProbCplex, 198, 206
readCopySolCplex, 207
refineConflictCplex, 208
refineConflictExtCplex, 209
refineMIPstartConflictCplex, 210
refineMIPstartConflictExtCplex, 211
return_codeCplex, 83, 212
rhsSaCplex, 213
setDb1ParmCplex, 214
setDefaultParmCplex, 215
setIntParmCplex, 216, 219
setLogFileCplex, 131, 217
setLogFileNameCplex, 218
setLongParmCplex, 219
setObjDirCplex, 220
setStrParmCplex, 221
setTerminateCplex, 44, 95, 201, 222
siftoptCplex, 223
solnInfoCplex, 15, 84, 97, 99, 187, 188, 190, 191, 200, 202, 224, 225
solutionCplex, 15, 84, 97, 99, 187, 188, 190, 191, 200, 202, 224, 225
solWriteCplex, 226
status_codeCplex, 83, 227
summary, cplexPtr-method (cplexPtr-class), 84
tightenBndsCplex, 32, 228
tuneParmCplex, 229
unscaleProbCplex, 230
writeMIPstartsCplex, 231
writeParmCplex, 232
writeProbCplex, 233