Package ‘cplexAPI’

September 21, 2020

Type Package
Title R Interface to C API of IBM ILOG CPLEX
Version 1.4.0
Date 2020-09-17
Depends R (>= 2.6.0)
Imports methods
Description This is the R Interface to the C API of IBM ILOG CPLEX. It necessarily depends 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 2020-09-21 16:20:11 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
R topics documented:

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
cCloneProbCPLEX .......................................................... 45
closeEnvCPLEX ............................................................. 46
closeFileCPLEX ............................................................ 47
closeProbCPLEX ............................................................ 48
cLpWriteCPLEX ............................................................. 49
completeLPCPLEX ........................................................ 50
copyBaseCPLEX ............................................................ 51
copyColTypeCPLEX ........................................................ 52
copyLpCPLEX ............................................................... 53
copyLpwNamesCPLEX ..................................................... 54
copyObjNameCPLEX ........................................................ 55
copyOrderCPLEX ........................................................... 56
copyQPsepCPLEX ........................................................... 57
copyQuadCPLEX ............................................................. 58
copyStartCPLEX ............................................................ 59
cplexConstants ............................................................. 60
cplexError-class ........................................................... 82
cplexPtr-class ............................................................. 83
R topics documented:

delColsCPLEX ........................................ 84
delFpDestCPLEX ...................................... 85
delIndConstrsCPLEX ................................. 86
delMIPstartsCPLEX ................................... 87
delNamesCPLEX ....................................... 88
delProbCPLEX ......................................... 89
delQConstrsCPLEX .................................... 90
delRowsCPLEX ........................................ 91
delSetColsCPLEX ...................................... 92
delSetRowsCPLEX ..................................... 93
delTerminateCPLEX .................................... 94
disconnectChannelCPLEX ......................... 95
dualoptCPLEX ......................................... 96
dualWriteCPLEX ....................................... 97
feasOptCPLEX ......................................... 98
fileputCPLEX .......................................... 99
flushChannelCPLEX ................................... 100
flushStdChannelsCPLEX ............................. 101
freePresolveCPLEX .................................... 102
getBaseCPLEX ......................................... 103
getBestObjValCPLEX .................................. 104
getChannelsCPLEX .................................... 105
getChgParmCPLEX ..................................... 106
getCoefCPLEX ......................................... 107
getColIndexCPLEX .................................... 108
getColInfeasCPLEX ................................... 109
getColNameCPLEX ..................................... 110
getColsCPLEX ......................................... 111
getColTypeCPLEX ...................................... 112
getConflictCPLEX ..................................... 113
getConflictExtCPLEX ................................ 114
cutoffCPLEX .......................................... 115
dblParmCPLEX ......................................... 116
dblQualCPLEX ......................................... 117
dbsCntCPLEX .......................................... 118
djCPLEX ............................................... 119
gErrorStrCPLEX ....................................... 120
gGradCPLEX ............................................ 121
gIndConstrCPLEX ..................................... 122
dblParmCPLEX ......................................... 123
dblParmCPLEX ......................................... 124
dblParmCPLEX ......................................... 125
dblParmCPLEX ......................................... 126
dblParmCPLEX ......................................... 127
dblParmCPLEX ......................................... 128
gItCntCPLEX ........................................... 129
gLogFileCPLEX ......................................... 130
gLogFileCPLEX ......................................... 131
### R topics documented:

- `getLongParmCplex` .................................................. 132
- `getLowBndsIdsCplex` ............................................... 133
- `getLowerBndsCplex` ................................................ 134
- `getMethodCplex` ...................................................... 135
- `getMIPrelGapCplex` ................................................ 136
- `getMIPstartIndexCplex` ............................................ 137
- `getMIPstartNameCplex` ............................................ 138
- `getMIPstartsCplex` ................................................ 139
- `getNumColsCplex` ................................................... 140
- `getNumMIPstartsCplex` ............................................ 141
- `getNumNnzCplex` .................................................... 142
- `getNumQConstrsCplex` ............................................ 143
- `getNumQPnzCplex` .................................................. 144
- `getNumQuadCplex` ................................................... 145
- `getNumRowsCplex` ................................................... 146
- `getObjCplex` .......................................................... 147
- `getObjDirCplex` ...................................................... 148
- `getObjNameCplex` .................................................... 149
- `getObjOffsetCplex` ................................................ 150
- `getObjValCplex` ...................................................... 151
- `getOrderCplex` ........................................................ 152
- `getParmHierNameCplex` ............................................ 153
- `getParmNameCplex` .................................................. 154
- `getParmNumCplex` ................................................... 155
- `getParmTypeCplex` .................................................. 156
- `getParmValCplex` .................................................... 157
- `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
- `getRowNameCplex` .................................................... 171
- `getRowsCplex` ........................................................ 172
- `getSenseCplex` ....................................................... 173
- `getSiftItCntCplex` .................................................. 174
- `getSiftPost1CntCplex` ............................................. 175
- `getSlackCplex` ....................................................... 176
- `getStatCplex` ........................................................... 177
- `getStatStrCplex` ..................................................... 178
topics documented:

getStrParmCPLEX ........................................... 179
getSubMethodCPLEX ........................................ 180
getSubStatCPLEX ........................................... 181
getAddressCPLEX .......................................... 182
getUppBndsIdsCPLEX ..................................... 183
getUpperBndsCPLEX ....................................... 184
getVersionCPLEX ........................................... 185
hybbaroptCPLEX .......................................... 185
hybnetoptCPLEX .......................................... 186
initProbCPLEX ............................................. 187
lpoptCPLEX ................................................ 188
mipoptCPLEX .............................................. 189
newColsCPLEX ............................................. 190
newRowsCPLEX ............................................. 191
objcSA CPLEX ............................................. 192
openEnvCPLEX ............................................. 193
openFileCPLEX ............................................. 194
openProbCPLEX ............................................ 195
ordWriteCPLEX ............................................ 196
preslvWriteCPLEX ........................................ 197
presolveCPLEX ............................................ 198
primoptCPLEX ............................................. 199
printTerminateCPLEX .................................... 200
qpoptCPLEX ................................................ 200
readCopyBaseCPLEX ....................................... 201
readCopyMIPstartsCPLEX ................................. 202
readCopyOrderCPLEX .................................... 203
readCopyParmCPLEX ..................................... 204
readCopyProbCPLEX ..................................... 205
readCopySolCPLEX ....................................... 206
refineConflictCPLEX .................................... 207
refineConflictExtCPLEX ................................ 208
refineMIPstartConflictCPLEX ........................... 209
refineMIPstartConflictExtCPLEX ....................... 210
return_codeCPLEX ....................................... 211
rhsSA CPLEX .............................................. 212
setDbiParamCPLEX ........................................ 213
setDefaultParmCPLEX ................................... 214
setIntParmCPLEX ......................................... 215
setLogFileCPLEX ......................................... 216
setLogFileNameCPLEX ................................... 217
setLongParmCPLEX ....................................... 218
setObjDirCPLEX .......................................... 219
setStrParmCPLEX ......................................... 220
setTerminateCPLEX ...................................... 221
siftoptCPLEX ............................................. 222
solnInfoCPLEX ............................................ 223
solutionCPLEX ............................................. 224
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
nc <- 3
nr <- 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, allocated 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

See Also

checkAddColsCPLEX, addRowsCPLEX

---

**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

addIndConstrCPLEX(env, lp, indvar, complemented, nzcnt, rhs, sense, linind, linval, indname = 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>indvar</td>
<td>The binary variable that acts as the indicator for this constraint.</td>
</tr>
<tr>
<td>complemented</td>
<td>A Boolean value that specifies whether the indicator variable is complemented.</td>
</tr>
<tr>
<td>nzcnt</td>
<td>An integer that specifies the number of nonzero coefficients in the linear portion of the indicator constraint.</td>
</tr>
<tr>
<td>rhs</td>
<td>The righthand side value for the linear portion of the indicator constraint.</td>
</tr>
<tr>
<td>sense</td>
<td>The sense of the linear portion of the indicator constraint.</td>
</tr>
<tr>
<td>linind</td>
<td>A vector that with linval defines the linear portion of the indicator constraint.</td>
</tr>
<tr>
<td>linval</td>
<td>A vector that with linind defines the linear portion of the indicator constraint.</td>
</tr>
<tr>
<td>indname</td>
<td>The name of the constraint to be added (optional).</td>
</tr>
</tbody>
</table>

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

```r
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**: Righthand 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.

References

The IBM ILOG CPLEX home page at https://www.ibm.com/support/knowledgecenter/SSSA5P.
**addRowsCPLEX**  
*Add Constraints to a Specified CPLEX Problem Object*

**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 CPXaddrows. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

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

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>env</code></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><code>lp</code></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><code>ncols</code></td>
<td>Number of new columns in the constraints being added to the constraint matrix.</td>
</tr>
<tr>
<td><code>nrows</code></td>
<td>Number of rows.</td>
</tr>
<tr>
<td><code>nnz</code></td>
<td>Number of nonzero constraint coefficients.</td>
</tr>
<tr>
<td><code>matbeg</code></td>
<td>An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><code>matind</code></td>
<td>An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><code>matval</code></td>
<td>An array used with rmatind and rmatval to define the rows to be added. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td><code>rhs</code></td>
<td>Righthand side term for each constraint to be added.</td>
</tr>
<tr>
<td><code>sense</code></td>
<td>Sense of each constraint to be added.</td>
</tr>
<tr>
<td><code>cnames</code></td>
<td>Names of the new columns.</td>
</tr>
<tr>
<td><code>rnames</code></td>
<td>Names of the new rows.</td>
</tr>
</tbody>
</table>
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

checkAddRowsCPLEX, addColsCPLEX, copyLpCPLEX, chgRngValCPLEX

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

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 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

```r
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>
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 CPXcheckaddcols.

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>
checkAddRowsCPLEX

References

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

See Also

addColsCPLEX

---

Description

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

Usage

```r
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.
checkChgCoefListCPLEX

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

---

checkChgCoefListCPLEX  Validate Arguments of the Corresponding chgCoefListCPLEX Routine

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, 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 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

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

checkCopyLpwNamesCPLEX

Validate Arguments of the Corresponding copyLpwNamesCPLEX Routine

Description

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

Usage

checkCopyLpwNamesCPLEX(env, lp, nCols, nRows, lpdir, 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.
lpdir Single integer value that specifies whether the problem is a minimization or maximization problem.
The objective function coefficients.

The righthand side values for each constraint in the constraint matrix.

The sense of each constraint in the constraint matrix.

Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

Array that defines the constraint matrix. Consult the IBM ILOG CPLEX documentation for more detailed information.

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.

Containing the lower bound on each of the variables.

Containing the range value of each ranged constraint.

Names of the matrix columns or, equivalently, the variable names.

Names of the matrix rows or, equivalently, the constraint names.

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

<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>qsepvec</td>
<td>A vector containing the quadratic coefficients.</td>
</tr>
</tbody>
</table>

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
Description

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

Usage

```
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
checkValsCPLEX  

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

checkValsCPLEX(env, lp, nval, rind = NULL, cind = NULL, val = 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.

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.
chgBndsCPLEX

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 CPXchgbds. Consult the IBM ILOG CPLEX documentation for more detailed information.

Usage
chgBndsCPLEX(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 chgBnds 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
getLowerBndsCPLEX, getUpperBndsCPLEX
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

ggetNumRowsCPLEX, getNumColsCPLEX, chgObjCPLEX, chgRhsCPLEX, chgRngValCPLEX
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**

```r
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**

### 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

```r
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


### 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, xctype)

Arguments

denv

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

dlp

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

dncols

Number of bounds to be changed.

dind

Indices of bounds to be changed.

dxctype

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.
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

chgMIPstartsCPLEX(env, lp, mcnt, mipstartindices, nzcnt, beg, varindices, values, effortlevel)

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 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 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.

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.

| chgNameCPLEX | 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
chgNameCPLEX(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

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

Change the Name of the Current Problem.

Description

Low level interface function to the IBM ILOG CPLEX function CPXchgpro name. 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 CPXchgpro name.

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.
chgProbTypeCPLEX  
*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
cchgProbTypeCPLEX(env, lp, ptype)
```

**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.

- `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**

`getProbTypeCPLEX`, `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

chgQPcoefCPLEX(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 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

chgRngValCPLEX

Change Range Coefficients

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 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**

chgTerminateCPLEX  
*Change Termination Signal*

**Description**
The function `chgTerminateCPLEX` changes termination signal.

**Usage**
```r
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**

**See Also**
- `setTerminateCPLEX`, `delTerminateCPLEX`, `printTerminateCPLEX`

---

cleanupCoefCPLEX  
*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**
```r
cleanupCoefCPLEX(env, lp, eps)
```
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


---

**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`. 
Value

If successful a pointer to the new CPLEX problem object as returned by `initProbCPLEX` (an object 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.

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-duesseldor.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

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


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**

```r
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 CPXfreeprob.

**Value**

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

**Author(s)**

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

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

**References**


**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

cLpWriteCPLEX(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 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 https://www.ibm.com/support/knowledgecenter/SSSA5P.
completelpCPLEX  Manage Modification Steps Closely

Description

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

Usage

completelpCPLEX(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 completelpCPLEX which calls the CPLEX function CPXcompletelp.

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

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

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

copyColTypeCPLEX(env, lp, xtype)

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>xtype</td>
<td>A vector containing the type of each column in the constraint matrix.</td>
</tr>
</tbody>
</table>

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

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

*Copy Data Defining an LP Problem to a CPLEX Problem Object.*

**Description**

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

**Usage**

```r
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.

copyLpwNamesCPLEX Copy Data Defining an LP Problem to a CPLEX Problem Object.

Description

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

Usage

copyLpwNamesCPLEX(env, lp, nCols, nRows, lpdire, 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.

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.

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.

copyObjNameCPLEX 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

copyObjNameCPLEX(env, lp, oname)

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.

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.

---

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.
copyQuadCPLEX Copy a Quadratic Objective Matrix Q When Q is not Diagonal.

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

Usage

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

Provides Starting Information for Use in a Subsequent Call to a Simplex Optimization Routine.

Description

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

Usage

```r
copyStartCPLEX(env, lp, cstat = NULL, rstat = NULL, cprim = NULL, rprim = NULL, cdual = NULL, rdual = 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.
- `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>Value</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 $y = ax + b$)
- CPX_PRECOL_OTHER -5 (cannot be expressed by a linear combination of active variables in the presolved model; crushing will fail if it has to touch such a variable)
- 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
Dual simplex pricing algorithm

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_PPRIIND_STEEP</td>
<td>2</td>
</tr>
<tr>
<td>CPX_PPRIIND_STEEPQSTART</td>
<td>3</td>
</tr>
<tr>
<td>CPX_PPRIIND_FULL</td>
<td>4</td>
</tr>
</tbody>
</table>

PARALLELMODE values

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_PARALLEL_DETERMINISTIC</td>
<td>1</td>
</tr>
<tr>
<td>CPX_PARALLEL_AUTO</td>
<td>0</td>
</tr>
<tr>
<td>CPX_PARALLEL_OPPORTUNISTIC</td>
<td>-1</td>
</tr>
</tbody>
</table>

Values for CPX_PARAM_WRITELEVEL

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_WRITELEVEL_AUTO</td>
<td>0</td>
</tr>
<tr>
<td>CPX_WRITELEVEL_ALLVARS</td>
<td>1</td>
</tr>
<tr>
<td>CPX_WRITELEVEL_DISCRETEVARS</td>
<td>2</td>
</tr>
<tr>
<td>CPX_WRITELEVEL_NONZEROVARS</td>
<td>3</td>
</tr>
<tr>
<td>CPX_WRITELEVEL_NONZERODISCRETEVARS</td>
<td>4</td>
</tr>
</tbody>
</table>

Values for CPX_PARAM_SOLUTIONTARGET

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_SOLUTIONTARGET_AUTO</td>
<td>0</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_OPTIMALCONVEX</td>
<td>1</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_FIRSTORDER</td>
<td>2</td>
</tr>
<tr>
<td>CPX_SOLUTIONTARGET_OPTIMALGLOBAL</td>
<td>3</td>
</tr>
</tbody>
</table>
LP/QP solution algorithms

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

```
CPX_ALG_NONE        -1
CPX_ALG_AUTOMATIC   0
CPX_ALG_PRIMAL      1
CPX_ALG_DUAL        2
CPX_ALG_NET         3
CPX_ALG_BARRIER     4
CPX_ALG_SIFTING     5
CPX_ALG_CONCURRENT  6
CPX_ALG_BAROPT      7
CPX_ALG_PIVOTIN     8
CPX_ALG_PIVOTOUT    9
CPX_ALG_PIVOT       10
CPX_ALG_FEASOPT     11
CPX_ALG_MIP         12
CPX_ALG_ROBUST      13
```

Basis status values

```
CPX_AT_LOWER        0
CPX_BASIC           1
CPX_AT_UPPER        2
CPX_FREE_SUPER      3
```

Variable types for ctype array

```
CPX_CONTINUOUS     "C"
CPX_BINARY         "B"
CPX_INTEGER        "I"
CPX_SEMICONT       "S"
CPX_SEMIINT        "N"
```

PREREDUCE settings
cplexConstants

CPX_PREREDUCE_PRIMALANDDUAL 3
CPX_PREREDUCE_DUALONLY 2
CPX_PREREDUCE_PRIMALONLY 1
CPX_PREREDUCE_NOPRIMALORDUAL 0

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_DET_TIME_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 Constants

- CPXPROB_FIXEDMIQP 8
- CPXPROB_NODEQP 9
- CPXPROB_QCP 10
- CPXPROB_MIQCP 11
- CPXPROB_NODEQCP 12

### CPLEX Parameter numbers

- CPX_PARAM_ADVIND 1001
- CPX_PARAM_AGGFILL 1002
- CPX_PARAM_AGGIND 1003
- CPX_PARAM_BASINTERVAL 1004
- CPX_PARAM_CFILEMUL 1005
- CPX_PARAM_CLOCKTYPE 1006
- CPX_PARAM_CRAIND 1007
- CPX_PARAM_DEPIND 1008
- CPX_PARAM_DPRIIND 1009
- CPX_PARAM_PRICELIM 1010
- CPX_PARAM_EPMRK 1013
- CPX_PARAM_EPOPT 1014
- CPX_PARAM_EPPER 1015
- CPX_PARAM_EPRHS 1016
- CPX_PARAM_FASTMIP 1017
- CPX_PARAM_SIMDISPLAY 1019
- CPX_PARAM_ITLIM 1020
- CPX_PARAM_ROWREADLIM 1021
- CPX_PARAM_NETFIND 1022
- CPX_PARAM_COLREADLIM 1023
- CPX_PARAM_NZREADLIM 1024
- CPX_PARAM_OBJULLIM 1025
- CPX_PARAM_OBJJULIM 1026
- CPX_PARAM_PERIND 1027
- CPX_PARAM_PERLIM 1028
- CPX_PARAM_PPRIIND 1029
- CPX_PARAM_PREIND 1030
- CPX_PARAM_REINV 1031
- CPX_PARAM_REVERSEIND 1032
- CPX_PARAM_RFFILEMUL 1033
- CPX_PARAM_SCAIND 1034
- CPX_PARAM_SCRIND 1035
- CPX_PARAM_SINGLIM 1037
- CPX_PARAM_SINGTOL 1038
- CPX_PARAM_TILIM 1039
- CPX_PARAM_XXIND 1041
- CPX_PARAM_PREDUAL 1044
- CPX_PARAM_EPOPT_H 1049
- CPX_PARAM_EPRHS_H 1050
<table>
<thead>
<tr>
<th>Constant Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_PARAM_PREPASS</td>
<td>1052</td>
</tr>
<tr>
<td>CPX_PARAM_DATACHECK</td>
<td>1056</td>
</tr>
<tr>
<td>CPX_PARAM_REDUCE</td>
<td>1057</td>
</tr>
<tr>
<td>CPX_PARAM_PRELINEAR</td>
<td>1058</td>
</tr>
<tr>
<td>CPX_PARAM_LPMETHOD</td>
<td>1062</td>
</tr>
<tr>
<td>CPX_PARAM_QPMETHOD</td>
<td>1063</td>
</tr>
<tr>
<td>CPX_PARAM_WORKDIR</td>
<td>1064</td>
</tr>
<tr>
<td>CPX_PARAM_WORKMEM</td>
<td>1065</td>
</tr>
<tr>
<td>CPX_PARAM_THREADS</td>
<td>1067</td>
</tr>
<tr>
<td>CPX_PARAM_CONFLICTDISPLAY</td>
<td>1074</td>
</tr>
<tr>
<td>CPX_PARAM_SIFTDISPLAY</td>
<td>1076</td>
</tr>
<tr>
<td>CPX_PARAM_SIFTAILG</td>
<td>1077</td>
</tr>
<tr>
<td>CPX_PARAM_SIFTITLIM</td>
<td>1078</td>
</tr>
<tr>
<td>CPX_PARAM_MPSLONGNUM</td>
<td>1081</td>
</tr>
<tr>
<td>CPX_PARAM_MEMORYEMPHASIS</td>
<td>1082</td>
</tr>
<tr>
<td>CPX_PARAM_NUMERICALEMPHASIS</td>
<td>1083</td>
</tr>
<tr>
<td>CPX_PARAM_FEASOPTMODE</td>
<td>1084</td>
</tr>
<tr>
<td>CPX_PARAM_PARALLELMODE</td>
<td>1109</td>
</tr>
<tr>
<td>CPX_PARAM_TUNINGMEASURE</td>
<td>1110</td>
</tr>
<tr>
<td>CPX_PARAM_TUNINGREPEAT</td>
<td>1111</td>
</tr>
<tr>
<td>CPX_PARAM_TUNINGTILIM</td>
<td>1112</td>
</tr>
<tr>
<td>CPX_PARAM_TUNINGDISPLAY</td>
<td>1113</td>
</tr>
<tr>
<td>CPX_PARAM_WRITELEVEL</td>
<td>1114</td>
</tr>
<tr>
<td>CPX_PARAM_RANDOMSEED</td>
<td>1124</td>
</tr>
<tr>
<td>CPX_PARAM_DETTILIM</td>
<td>1127</td>
</tr>
<tr>
<td>CPX_PARAM_FILEENCODING</td>
<td>1129</td>
</tr>
<tr>
<td>CPX_PARAM_APIENCODING</td>
<td>1130</td>
</tr>
<tr>
<td>CPX_PARAM_SOLUTIONTARGET</td>
<td>1131</td>
</tr>
<tr>
<td>CPX_PARAM_CLONELOG</td>
<td>1132</td>
</tr>
<tr>
<td>CPX_PARAM_TUNINGDETTILIM</td>
<td>1139</td>
</tr>
<tr>
<td>CPX_PARAM_ALL_MIN</td>
<td>1000</td>
</tr>
<tr>
<td>CPX_PARAM_ALL_MAX</td>
<td>6000</td>
</tr>
</tbody>
</table>

**Values for** `CPX_PARAM_TUNINGMEASURE`

- CPX_TUNE_AVERAGE: 1
- CPX_TUNE_MINMAX: 2

**Values for incomplete tuning**

- CPX_TUNE_ABORT: 1
**cplexConstants**

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPX_TUNE_TILIM</td>
<td>2</td>
</tr>
<tr>
<td>CPX_TUNE_DETTILIM</td>
<td>3</td>
</tr>
</tbody>
</table>

### Quality query identifiers

<table>
<thead>
<tr>
<th>Quality Query 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_SUM_COMP_SLACK</td>
<td>20</td>
</tr>
<tr>
<td>CPX_MAX_X</td>
<td>21</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_X</td>
<td>22</td>
</tr>
<tr>
<td>CPX_MAX_PI</td>
<td>23</td>
</tr>
<tr>
<td>CPX_MAX_SCALED_PI</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>
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_BARQCPEPCOMP 3020

Optimizing Problems

CPX_BARORDER_AUTO 0
CPX_BARORDER_AMD 1
CPX_BARORDER_AMF 2
CPX_BARORDER_ND 3

MIP emphasis settings

CPX_MIPEMPHASIS_BALANCED 0
CPX_MIPEMPHASIS_FEASIBILITY 1
CPX_MIPEMPHASIS_OPTIMALITY 2
CPX_MIPEMPHASIS_BESTBOUND 3
CPX_MIPEMPHASIS_HIDDENFEAS 4

Values for sostype and branch type

CPX_TYPE_VAR "0"
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

CplexConstants
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

\[
\begin{align*}
\text{CPX_USECUT_FORCE} & \quad 0 \\
\text{CPX_USECUT_PURGE} & \quad 1 \\
\text{CPX_USECUT_FILTER} & \quad 2 \\
\end{align*}
\]

For \text{CPXgetnodeintfeas}

\[
\begin{align*}
\text{CPX_INTEGER_FEASIBLE} & \quad 0 \\
\text{CPX_INTEGER_INFEASIBLE} & \quad 1 \\
\end{align*}
\]
cplexConstants

CPX_IMPLIED_INTEGER_FEASIBLE  2

MIP Parameter numbers

CPX_PARAM_BRDIR 2001
CPX_PARAM_BTTOL 2002
CPX_PARAM_CLIQUES 2003
CPX_PARAM_COEREDIND 2004
CPX_PARAM_COVERS 2005
CPX_PARAM_CUTLO 2006
CPX_PARAM_CUTUP 2007
CPX_PARAM_EPAGAP 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_OBJDIF 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_BNDSTRENIND 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_STRONGCANDLIM 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_SOLNPOLOREPLACE`
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

CPX_CON_LOWER_BOUND 1
CPX_CON_UPPER_BOUND 2
CPX_CON_LINEAR 3
CPX_CON_QUADRATIC 4
CPX_CON_SOS 5
CPX_CON_INDICATOR 6

internal types

CPX_CON_MINEXPR 7
CPX_CON_MAXEXPR 8
CPX_CON_PWL 9
CPX_CON_ABS 9
CPX_CON_DISJCST 10
CPX_CON_INDDISJCST 11
CPX_CON_SETVAR 12
CPX_CON_SETVARMEMBER 13
CPX_CON_SETVARCARD 14
CPX_CON_SETVARSUM 15
CPX_CON_SETVARMIN 16
CPX_CON_SETVARMAX 17
CPX_CON_SETVARSUBSET 18
CPX_CON_SETVARDOMAIN 19
CPX_CON_SETVARUNION 20
CPX_CON_SETVARINTERSECTION 21
CPX_CON_SETVARNULLINTERSECT 22
CPX_CON_SETVARINTERSECT 23
CPX_CON_SETVAREQ 24
CPX_CON_SETVAREQ 25
CPX_CON_SETVARNEQCST 26
CPX_CON_LAST_CONTYPE 27
Network parameters

- CPX_PARAM_NETITLIM 5001
- CPX_PARAM_NETEOPT 5002
- CPX_PARAM_NETEPRHS 5003
- CPX_PARAM_NETPPRIIND 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
cplexConstants

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_SUBPROB_SOLVE 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
CPXPARAM_BARRIER_QCPConvergeTol 3020
CPXPARAM_BARRIER_StartAlg 3017
CPXPARAM_ClockType 1006
CPXPARAM_Conflict_Display 1074
CPXPARAM_DetTimeLimit 1127
CPXPARAM_DistMIP_Rampup_DetTimeLimit 2164
CPXPARAM_DistMIP_Rampup_Duration 2163
CPXPARAM_DistMIP_Rampup_TimeLimit 2165
CPXPARAM_Emphasis_Memory 1082
CPXPARAM_Emphasis_MIP 2058
CPXPARAM_Emphasis_Numerical 1083
CPXPARAM_Feasopt_Mode 1084
CPXPARAM_Feasopt_Tolerance 2073
CPXPARAM_LPMethod 1062
CPXPARAM_MIP_Cuts_Cliques 2003
CPXPARAM_MIP_Cuts_Covers 2005
CPXPARAM_MIP_Cuts_Disjunctive 2053
CPXPARAM_MIP_Cuts_FlowCovers 2040
CPXPARAM_MIP_Cuts_Gomory 2049
CPXPARAM_MIP_Cuts_GUBCovers 2044
CPXPARAM_MIP_Cuts_Implied 2041
CPXPARAM_MIP_Cuts_LiftProj 2152
CPXPARAM_MIP_Cuts_MCFCut 2134
CPXPARAM_MIP_Cuts_MIRCut 2052
CPXPARAM_MIP_Cuts_PathCut 2051
CPXPARAM_MIP_Cuts_ZeroHalfCut 2111
CPXPARAM_MIP_Display 2012
CPXPARAM_MIP_Interval 2013
CPXPARAM_MIP_Limits_AggForCut 2054
CPXPARAM_MIP_Limits_AuxRootThreads 2139
CPXPARAM_MIP_Limits_CutPasses 2056
CPXPARAM_MIP_Limits_CutsFactor 2033
CPXPARAM_MIP_Limits_EachCutLimit 2102
CPXPARAM_MIP_Limits_GomoryCand 2048
CPXPARAM_MIP_Limits_GomoryPass 2050
CPXPARAM_MIP_Limits_Nodes 2017
CPXPARAM_MIP_Limits_PolishTime 2066
CPXPARAM_MIP_Limits_Populate 2108
CPXPARAM_MIP_Limits_ProbeDetTime 2150
CPXPARAM_MIP_Limits_ProbeTime 2065
CPXPARAM_MIP_Limits_RepairTries 2067
CPXPARAM_MIP_Limits_Solutions 2015
CPXPARAM_MIP_Limits_StrongCand 2045
CPXPARAM_MIP_Limits_StrongIt 2046
CPXPARAM_MIP_Limits_SubMIPNodeLim 2062
CPXPARAM_MIP_Limits_TreeMemory 2027
CPXPARAM_MIP_OrderType 2032
CPXPARAM_MIP_PolishAfter_AbsMIPGap 2126
CPXPARAM_MIP_PolishAfter_DetTime 2151
CPXPARAM_MIP_PolishAfter_MIPGap 2127
CPXPARAM_MIP_PolishAfter_Nodes 2128
CPXPARAM_MIP_PolishAfter_Solutions 2129
CPXPARAM_MIP_PolishAfter_Time 2130
CPXPARAM_MIP_Pool_AbsGap 2106
CPXPARAM_MIP_Pool_Capacity 2103
CPXPARAM_MIP_Pool_Intensity 2107
CPXPARAM_MIP_Pool_RelGap 2105
CPXPARAM_MIP_Pool_Replace 2104
CPXPARAM_MIP_Strategy_Backtrack 2002
CPXPARAM_MIP_Strategy_BBInterval 2039
CPXPARAM_MIP_Strategy_Branch 2001
CPXPARAM_MIP_Strategy_CallbackReducedLP 2055
CPXPARAM_MIP_Strategy_Dive 2060
CPXPARAM_MIP_Strategy_File 2016
CPXPARAM_MIP_Strategy_FPHeur 2098
CPXPARAM_MIP_Strategy_HeuristicFreq 2031
CPXPARAM_MIP_Strategy_KappaStats 2137
CPXPARAM_MIP_Strategy_LBHeur 2063
CPXPARAM_MIP_Strategy_MIQCPStrat 2110
CPXPARAM_MIP_Strategy_NodeSelect 2018
CPXPARAM_MIP_Strategy_Order 2020
CPXPARAM_MIP_Strategy_PresolveNode 2037
CPXPARAM_MIP_Strategy_Probe 2042
CPXPARAM_MIP_Strategy_RINSHeur 2061
CPXPARAM_MIP_Strategy_Search 2109
CPXPARAM_MIP_Strategy_StartAlgorithm 2025
CPXPARAM_MIP_Strategy_SubAlgorithm 2026
CPXPARAM_MIP_Strategy_VariableSelect 2028
CPXPARAM_MIP_Tolerances_AbsMIPGap 2008
CPXPARAM_MIP_Tolerances_Integrality 2010
CPXPARAM_MIP_Tolerances_LowerCutoff 2006
CPXPARAM_MIP_Tolerances_MIPGap 2009
CPXPARAM_MIP_Tolerances_ObjDifference 2019
CPXPARAM_MIP_Tolerances_RelObjDifference 2022
CPXPARAM_MIP_Tolerances_UpperCutoff 2007
CPXPARAM_Network_Display 5005
CPXPARAM_Network_Iterations 5001
CPXPARAM_Network_NetFind 1022
CPXPARAM_Network_Pricing 5004
CPXPARAM_Network_Tolerances_Feasibility 5003
CPXPARAM_Network_Tolerances_Optimality 5002
CPXPARAM_Output_CloneLog 1132
CPXPARAM_Output_IntSolFilePrefix 2143
CPXPARAM_Output_MPSLong 1081
CPXPARAM_Output_WriteLevel 1114
CPXPARAM_Parallel 1109
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<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>
cplexConstants

CPXPARAM_Tune_DetTimeLimit 1139
CPXPARAM_Tune_Display 1113
CPXPARAM_Tune_Measure 1110
CPXPARAM_Tune_Repeat 1111
CPXPARAM_Tune_TimeLimit 1112
CPXPARAM_WorkDir 1064
CPXPARAM_WorkMem 1065

new parameter names introduced in IBM ILOG CPLEX version 12.8.0

 Callable library miscellaneous routines

CPXPARAM_Record 1162
CPXPARAM_MIP_Strategy_SubMIPScale 2207
CPXPARAM_MIP_Strategy_SubMIPStartAlg 2205
CPXPARAM_MIP_Strategy_SubMIPSubAlg 2206
CPXPARAM_ParamDisplay 1163
CPX_PARAM_PARAMDISPLAY 1163
CPXMI_SAMECOEFF_ROW 1049
CPXMI_SAMECOEFF_COL 1050
CPXMI_SAMECOEFF_IND 1051
CPXMI_SAMECOEFF_QLIN 1052
CPXMI_SAMECOEFF_QUAD 1053
CPXMI_SAMECOEFF_LAZY 1054
CPXMI_SAMECOEFF_UCUT 1055
CPXMI_SAMECOEFF_RHS 1056
CPXMI_SAMECOEFF_OBJ 1057
CPX_CALLBACKCONTEXT_CANDIDATE 0x0020
CPX_CALLBACKCONTEXT_GLOBAL_PROGRESS 0x0010
CPX_CALLBACKCONTEXT_LOCAL_PROGRESS 0x0008
CPX_CALLBACKCONTEXT_RELAXATION 0x0040
CPX_CALLBACKCONTEXT_THREAD_DOWN 0x0004
CPX_CALLBACKCONTEXT_THREAD_UP 0x0002

new parameter names introduced in IBM ILOG CPLEX version 12.9.0

 Callable library miscellaneous routines

CPXPARAM_Read_WarningLimit 1157
CPXPARAM_MultiObjective_Display 1600
CPXPARAM_Preprocessing_Folding 1164

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

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

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

Methods

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

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

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

errnum<- 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

Class "cplexPtr"

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

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>
**delFpDestCplex**

References
---

---

**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

```r
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


### See Also

`addFpDestCplex`
Delete a Range of Indicator Constraints

Description

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

Usage

delIndConstrsCPLEX(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 indicator constraint to be deleted.
- **end**: An integer that specifies the numeric index of the last indicator constraint to be deleted.

Details

Interface to the C function delIndConstrs which calls the CPLEX function 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 https://www.ibm.com/support/knowledgecenter/SSSA5P.
Description

Low level interface function to the IBM ILOG CPLEX function CPXdelmipstarts. 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.

del 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 CPXdelmipstarts.

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.
\textbf{delNamesCPLEX} \hspace{2cm} \textit{Remove all Names Assigned to Rows and Columns}

\section*{Description}
Low level interface function to the IBM ILOG CPLEX function \texttt{CPXdelnames}. Consult the IBM ILOG CPLEX documentation for more detailed information.

\section*{Usage}
\begin{verbatim}
delNamesCPLEX(env, lp)
\end{verbatim}

\section*{Arguments}
\begin{itemize}
\item \texttt{env} \hspace{2cm} 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{2cm} An object of class \texttt{"cplexPtr"} as returned by \texttt{initProbCPLEX}. This is basically a pointer to an IBM ILOG CPLEX problem object.
\end{itemize}

\section*{Details}
Interface to the C function \texttt{delNames} which calls the CPLEX function \texttt{CPXdelnames}.

\section*{Value}
Zero if successful, otherwise nonzero.

\section*{Author(s)}
Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

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

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

Remove Specified CPLEX Problem Object

Description

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

Usage

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 CPXfreeprob.

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
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**

```
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**

delRowsCPLEX  Delete a Range of Rows

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.
**delSetColsCplex**  
*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**

```r
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 <geliudie@uni-duesseldorf.de>

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

**References**

**delSetRowsCPLEX**  
*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**
```r
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-duesseldorf.de>
Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**
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

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 CPXsetterminate.

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

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
dualoptCPLEX

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
dualoptCPLEX(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 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

solnInfoCPLEX, getStatCPLEX, solutionCPLEX
Write a Dual Formulation of the Current CPLEX Problem Object

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.
Description

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

Usage

feasOptCPLEX(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

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

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

```
fileputCPLEX(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

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

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

See Also

disconnectChannelCPLEX, flushChannelCPLEX, getChannelsCPLEX
**freePresolveCPLEX**  
*Free Presolved Problem From the LP Problem Object*

---

**Description**

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

**Usage**

```r
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**

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**

```r
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**

getChannelsCPLEX

Obtain Pointers to the Four Default Channels

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetchannels. 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 CPXgetchannels.

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 \hspace{1cm} \textit{Get Parameter Numbers for Parameters Which are Not Set at Their Default Values}

\textbf{Description}

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

\textbf{Usage}

getChgParmCPLEX(env)

\textbf{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.
\end{itemize}

\textbf{Details}

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

\textbf{Value}

A vector containing integer values (unique parameter identifiers) for parameters which are not set at their default values, 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}.

\textbf{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-duesseldorf.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

<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>cname</td>
<td>A column name to search for.</td>
</tr>
</tbody>
</table>

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

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

```r
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**

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

Accesses a Range of Columns of the Constraint Matrix

Description

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

Usage

getColsCPLEX(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 columns to be returned.
end An integer specifying the end of the range of columns to be returned.

Details

Interface to the C function getCols which calls the CPLEX function CPXgetcols.

Value

If successful a list is returned:

matbeg Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
matind Array that specifies the nonzero elements of the columns. Consult the IBM ILOG CPLEX documentation for more detailed information.
matval 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.
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

e   n   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

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  The index of the first group.

end  The index of the last group.

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

ggetCutoffCPLEX(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 getDb1Parm 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

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

See Also

cplexConstants
getDblQualCPLEX

Access 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

getDblQualCPLEX(env, lp, w)

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>w</td>
<td>An Integer specifying the quality value to be retrieved.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getDblQual 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**

```r
getDjCPLEX(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**

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

generateStrCPLEX(err, env = NULL)

Arguments

err The error code to be translated.

eEnv 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

generateStatsCPLEX
getGradCPLEX

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

getIndConstrCPLEX(env, lp, which)

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>which</td>
<td>An integer specifying which indicator constraint to return.</td>
</tr>
</tbody>
</table>

Details

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

Value

If successful a list is returned:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>indvar</td>
<td>Index of the binary indicator variable. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>complemented</td>
<td>Boolean value that specifies whether the indicator variable is complemented. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>rhs</td>
<td>Righthand side value of the linear portion of the indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>sense</td>
<td>Sense of the linear portion of the constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>linind</td>
<td>Variable indices of the entries of linval. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>linval</td>
<td>Coefficients of the linear portion of the specified indicator constraint. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
</tbody>
</table>

otherwise an instance of class "cplexError".
getInfoDblParmCPLEX

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>
getInfoIntParmCPLEX

### References


### 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

```r
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

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**

```
getInfoLongParmCPLEX(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>
References

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

See Also

getInfoIntParmCPLEX, 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

getInfoStrParmCPLEX(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

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**

```r
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**

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**

```r
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**


**See Also**

- `setLogFileCPLEX`
getLogFileNameCPLEX  

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

getLogFileNameCPLEX(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 getLogFileName 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 CPXgetlongparam. 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 CPXgetlongparam.

Value

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

Note

In order to get a 64 bit integer value from CPXgetlongparam, 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

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

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

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.

dead 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 <mayo.roettger@hhu.de>

References

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

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-duesseldor.f.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

gObjValCPLEX, 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

getMIPstartsCPLEX(env, lp, begin, end)

Arguments

eav 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.

dend 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 Gelius-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
g.getNumColsCPLEX(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 <code>openEnvCPLEX</code>. 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 <code>initProbCPLEX</code>. This is basically a pointer to an IBM ILOG CPLEX problem object.</td>
</tr>
</tbody>
</table>

**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**

getNumMIPstartsCPLEX

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

ggetNumMIPstartsCPLEX(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

**Description**

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

**Usage**

```r
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**

getNumQConstrsCPLEX  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

getNumQConstrsCPLEX(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 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

getNumQPnzCPLEX(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 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 functiongetNumQuad 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

```r
getNumRowsCPLEX(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 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/SSSA5P.
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

dep  
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

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

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

See Also

cplexConstants
getObjNameCPLEX

Access the Name of the Objective Row of a CPLEX Problem Object

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 CPXgetparamhiernamex. Consult the IBM ILOG CPLEX documentation for more detailed information. This function has no functionality for IBM ILOG CPLEX < 12.9.0 on, where CPXgetparamhiernamex 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 CPXgetparamhiernamex.

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

cgetParmNameCPLEX(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 getParamName 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 "cpexPtr" 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)
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.

getPiCPLEX

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

getPiCPLEX(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
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 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**


**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

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".
getQPcoefCPLEX

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.

getQPcoefCPLEX

Access the Quadratic Coefficient in the Matrix Q

Description

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

Usage

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 CPXgetqpcoeff.

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

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

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

getQuadCPLEX(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 columns to be returned.

dead 
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**

```r
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.

d_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**

```
getRowIndexCPLEX(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**

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**

ggetRowInfeasCPLEX(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>
Maintenance: Mayo Roettger <mayo.roettger@hhu.de>

**References**

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

Access a Range of Row Names

Description

Low level interface function to the IBM ILOG CPLEX function CPXgetrowname. 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

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

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

getSenseCPLEX(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 constraint senses to be returned.
- **end**: 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**  \hspace{1cm} *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
getsiftItCntCPLEX(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 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

**Description**

Low level interface function to the IBM ILOG CPLEX function `CPXgetsiftphase1cnt`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
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.
- **end**: 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

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

See Also

ggetErrorStrCPLEX
getStrParmCplex  
Obtain the Current Value of a CPLEX String Parameter

Description

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

Usage

getStrParmCplex(env, parm)

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>parm</td>
<td>Constant or reference number of the desired parameter.</td>
</tr>
</tbody>
</table>

Details

Interface to the C function getStrParm which calls the CPLEX function CPXgetstrparam.

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**

```r
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 &lt;geliudie@uni-duesseldorf.de&gt;  
Maintainer: Mayo Roettger &lt;mayo.roettger@hhu.de&gt;

**References**


**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 <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.
getTimeCPLEX

Get a Time Stamp

Description

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

Usage

getimeCPLEX(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 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, openFileCPLEX
getUppBndsIdsCPLEX

Retrieve Upper Bounds on Variables

Description

The function retrieves the upper bounds on specified variables.

Usage

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

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

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.
end    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**


---

**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”.

Description

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

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

---

lpoptCplex

Find a Solution to a Problem Using One of the CPLEX Linear Optimizers

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

**Author(s)**

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

**References**


**See Also**

solnInfoCPLEX, getStatCPLEX, solutionCPLEX

---

**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**

mipoptCPLEX(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 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**

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>

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

cplexConstants section “Variable types for ctype array”.

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

```r
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
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**

```r
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-duesseldorf.de>
- Maintainer: Mayo Roettger <mayo.roettger@hhu.de>

**References**


**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
openProbCPLEX  

Create new CPLEX Environment and New CPLEX Problem Object

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-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
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/knowledgecenter/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”.
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

```
primoptCPLEX(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

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

See Also

`solnInfoCPLEX`, `getStatCPLEX`, `solutionCPLEX`
printTerminateCPLEX  

*Print Termination Signal*

**Description**

The function `chgTerminateCPLEX` prints termination signal.

**Usage**

```r
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**

```r
qpoptCPLEX(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 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/knowledgecenter/SSSA5P.

See Also

solnInfoCPLEX, getStatCPLEX, solutionCPLEX

readCopyBaseCPLEX   Read Basis From a BAS File and Copy it Into a CPLEX Problem Object

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 openEnvCLEX. 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.
readCopyMIPstartsCPLEX

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.

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

<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>fname</td>
<td>Name of the file to read from.</td>
</tr>
</tbody>
</table>

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**

---

**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**
Description

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

Usage

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

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

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**

```r
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**

readCopySolCPLEX  Reads a Solution From a SOL Format File

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

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 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

<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>grpcnt</td>
<td>The number of constraint groups to be considered.</td>
</tr>
<tr>
<td>concnt</td>
<td>Length of arrays grpind and grptype.</td>
</tr>
<tr>
<td>grppref</td>
<td>Preferences for the groups.</td>
</tr>
<tr>
<td>grpbeg</td>
<td>The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>grpind</td>
<td>The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
<tr>
<td>grptype</td>
<td>The constraint indices. Consult the IBM ILOG CPLEX documentation for more detailed information.</td>
</tr>
</tbody>
</table>

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
Refine a Conflict in Order to Determine Why a Given MIP Start is Not Feasible

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
getConflictCPLEX
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

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

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

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

See Also

cplexConstants
rhsSaCPLEX

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

**References**

**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
**setIntParmCPLEX**  
*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**

```r
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**


**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

```r
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`
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 for IBM ILOG CPLEX < 12.8.0, where CPXsetlogfile was not included.

Usage

setLogFileNameCPLEX(env, filename = "cpx.log", mode = "w")

Arguments

evn
An object of class "cplexPtr" as returned by openEnvCPLEX. 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 CPXsetlogfile.

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, lpdir)

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.
- **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/knowledgecenter/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-duesseldorfe.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 `CPXsettermiate` with argument `terminate_p` set to `NULL`. Consult the IBM ILOG CPLEX documentation for more detailed information.

**Usage**

```r
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 `CPXsettermiate` 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 CPXsolution. 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 CPXsolution.

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 filname 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**


**See Also**

getProbTypeCPLEX, cplexConstants section “Problem Types”. 
status_codeCPLEX

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_codeCPLEX(env, code)

Arguments

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

code Status code from IBM ILOG CPLEX as returned by getStatCPLEX.

Value

A character string corresponding to the value of an IBM ILOG CPLEX status code as returned by getStatCPLEX.

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.

See Also

cplexConstants, getStatStrCPLEX
tightlyBndsCPLEX

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

```r
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


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

tuneParmCPLEX(env, lp,
  nIntP = 0, intP = NULL, intPv = NULL,
  nDb1P = 0, db1P = NULL, db1Pv = NULL,
  nStrP = 0, strP = NULL, strPv = 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.
  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.
  nDb1P        Number of double parameters to be fixed during tuning.
  db1P         Parameter numbers of the double parameters which remain fixed.
  db1Pv        Values for the parameters listed in db1P.
  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

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>
unscaleProbCPLEX

Remove Any Scaling Applied to the Resident Problem

Description

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

Usage

unscaleProbCPLEX(env, lp)

Arguments

ev 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 CPXunscaleprob.

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.
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

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 to write to.
begin An integer specifying the beginning of the range of MIP starts to be written.
end 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 <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.
**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**

```r
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**

Index

* optimize

addColsCPLEX, 7
addFpDestCPLEX, 9
addIndConstrsCPLEX, 10
addMIPstartsCPLEX, 11
addQConstrsCPLEX, 12
addRowsCPLEX, 13
baroptCPLEX, 14
baseWriteCPLEX, 15
basicPresolveCPLEX, 16
boundSaCPLEX, 17
checkAddColsCPLEX, 18
checkAddRowsCPLEX, 19
checkChgCoefListCPLEX, 20
checkCopyCoftypeCPLEX, 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
chgQPcoefsCPLEX, 39
chgRhsCPLEX, 40
chgRngValCPLEX, 41
chgRowNameCPLEX, 42
chgSenseCPLEX, 43
chgTerminateCPLEX, 44
cleanupCoefCPLEX, 44
closeEnvCPLEX, 46
closeFileCPLEX, 47
closeProbCPLEX, 48
clpWriteCPLEX, 49
completeLPCPLEX, 50
copyBaseCPLEX, 51
copyColTypeCPLEX, 52
copyLPCPLEX, 53
copyLpwNamesCPLEX, 54
copyObjNameCPLEX, 55
copyOrderCPLEX, 56
copyQPsepCPLEX, 57
copyQuadCPLEX, 58
copyStartCPLEX, 59
cplexAPI-package, 6
cplexConstants, 60
cplexError-class, 82
cplexPtr-class, 83
delColsCPLEX, 84
delFpDestCPLEX, 85
delIndConstrsCPLEX, 86
delMIPstartsCPLEX, 87
delNamesCPLEX, 88
delProbCPLEX, 89
delQConstrsCPLEX, 90
delRowsCPLEX, 91
delSetColsCPLEX, 92
delSetRowsCPLEX, 93
delTerminateCPLEX, 94
disconnectChannelCPLEX, 95
dualoptCPLEX, 96
dualWriteCPLEX, 97
feasOptCPLEX, 98
fileputCPLEX, 99
flushChannelCPLEX, 100
flushStdChannelsCPLEX, 101
freePresolveCPLEX, 102
getBaseCPLEX, 103
getBestObjValCPLEX, 104
INDEX

**package**
cplexAPI-package, 6

cplexAPI-package, 6

cplexAPI-package, 6

* package*
cplexAPI-package, 6

cplexAPI-package, 6

addColsCplex, 7, 14, 19
addFpDestCplex, 9, 85
addIndConstrCplex, 10
addMIPstartsCplex, 11
addQConstrCplex, 12
addRowsCplex, 9, 13, 20

baroptCplex, 14
baseWriteCplex, 15
basicPresolveCplex, 16

boundSaCplex, 17

checkAddColsCplex, 9, 18
checkAddRowsCplex, 14, 19
checkChgCoeflistCplex, 20
checkCopyColTypeCplex, 21
checkCopyLpCplex, 22
checkCopyLpWNamesCplex, 23
checkCopyQPsepCplex, 25
checkCopyQuadCplex, 26
checkValsCplex, 27
chgBndsCplex, 28, 32, 227
chgCoefCplex, 29, 39
chgCoefListCplex, 27, 30
chgColNameCplex, 31
chgColsBndsCplex, 32
chgColTypeCplex, 33
chgMIPstartsCplex, 34
chgNameCplex, 35
chgObjCplex, 29, 36
chgProbNameCplex, 37
chgProbTypeCplex, 38, 161
chgQPcoefficientCplex, 39
chgRhsCplex, 29, 40
chgRngValCplex, 14, 29, 41
chgRowNameCplex, 42
chgSenseCplex, 43
chgTerminateCplex, 44, 94, 200, 221
cleanupCoefficientCplex, 44
closeProbCplex, 45
closeEnvCplex, 46, 193
closeFileCplex, 47, 99, 194
closeProbCplex, 48, 195
cLPWriteCplex, 49
completeLPcplex, 50
constantsCplex (cplexConstants), 60
copyBaseCplex, 51
copyColTypeCplex, 22, 52
copyLpCplex, 14, 23, 53
copyLpWNamesCplex, 24, 54
copyObjNameCplex, 55
copyOrderCplex, 56
copyQPsepCplex, 25, 57

copyQuadCplex, 26, 58
copyStartCplex, 59
cplexConstants (cplexConstants), 60
cplexAPI (cplexAPI-package), 6
cplexAPI-package, 6
CPX_CON_SETVARNULLINTERSECT (cplexConstants), 60
CPX_CON_SETVARSUBSET (cplexConstants), 60
CPX_CON_SETVARSUM (cplexConstants), 60
CPX_CON_SETVARUNION (cplexConstants), 60
CPX_CON_SOS (cplexConstants), 60
CPX_CON_UPPER_BOUND (cplexConstants), 60
CPX_CONFLICT_EXCLUDED (cplexConstants), 60
CPX_CONFLICT_LB (cplexConstants), 60
CPX_CONFLICT_MEMBER (cplexConstants), 60
CPX_CONFLICT_POSSIBLE_LB (cplexConstants), 60
CPX_CONFLICT_POSSIBLE_MEMBER (cplexConstants), 60
CPX_CONFLICT_POSSIBLE_UB (cplexConstants), 60
CPX_CONFLICT_UB (cplexConstants), 60
CPX_CONTINUOUS (cplexConstants), 60
CPX_DPRIIND_AUTO (cplexConstants), 60
CPX_DPRIIND_DEVEX (cplexConstants), 60
CPX_DPRIIND_FULL (cplexConstants), 60
CPX_DPRIIND_FULL_STEEP (cplexConstants), 60
CPX_DPRIIND_STEEPQSTART (cplexConstants), 60
CPX_DUAL_OBJ (cplexConstants), 60
CPX_FEASOPT_MIN_INF (cplexConstants), 60
CPX_FEASOPT_MIN_QUAD (cplexConstants), 60
CPX_FEASOPT_MIN_SUM (cplexConstants), 60
CPX_FEASOPT_OPT_INF (cplexConstants), 60
CPX_FEASOPT_OPT_QUAD (cplexConstants), 60
CPX_FEASOPT_OPT_SUM (cplexConstants), 60
CPX_FREE_SUPER (cplexConstants), 60
CPX_IMPLIED_INTEGER_FEASIBLE (cplexConstants), 60
CPX_INFBOUND (cplexConstants), 60
CPX_INTEGER (cplexConstants), 60
CPX_INTEGER_FEASIBLE (cplexConstants), 60
CPX_INTEGER_INFEASIBLE (cplexConstants), 60
CPX_KAPPA (cplexConstants), 60
CPX_KAPPA_ATTENTION (cplexConstants), 60
CPX_KAPPA_IILLPOSED (cplexConstants), 60
CPX_KAPPA_MAX (cplexConstants), 60
CPX_KAPPA_STABLE (cplexConstants), 60
CPX_KAPPA_SUSPICIOUS (cplexConstants), 60
CPX_KAPPA_UNSTABLE (cplexConstants), 60
CPX_MAX (cplexConstants), 60
CPX_MAX_COMP_SLACK (cplexConstants), 60
CPX_MAX_DUAL_INFEAS (cplexConstants), 60
CPX_MAX_DUAL_RESIDUAL (cplexConstants), 60
CPX_MAX_INSLACK_INFEAS (cplexConstants), 60
CPX_MAX_INT_INFEAS (cplexConstants), 60
CPX_MAX_PI (cplexConstants), 60
CPX_MAX_PRIMAL_INFEAS (cplexConstants), 60
CPX_MAX_PRIMAL_RESIDUAL (cplexConstants), 60
CPX_MAX_PWLSLACK_INFEAS (cplexConstants), 60
CPX_MAX_QCPRIMAL_RESIDUAL (cplexConstants), 60
CPX_MAX_QCSLACK (cplexConstants), 60
CPX_MAX_QCSLACK_INFEAS (cplexConstants), 60
CPX_MAX_RED_COST (cplexConstants), 60
CPX_MAX_SCALING_DUAL_INFEAS (cplexConstants), 60
CPX_MAX_SCALING_DUAL_RESIDUAL (cplexConstants), 60
CPX_MAX_SCALING_PI (cplexConstants), 60
CPX_MAX_SCALING_PRIMAL_INFEAS (cplexConstants), 60
CPX_MAX_SCALING_PRIMAL_RESIDUAL (cplexConstants), 60
CPX_MAX_SCALING_RED_COST (cplexConstants), 60
CPX_MAX_SCALING_SLACK (cplexConstants), 60
CPX_MAX_SCALING_X (cplexConstants), 60
CPX_MAX_SLACK (cplexConstants), 60
CPX_MAX_X (cplexConstants), 60
CPX_MIN (cplexConstants), 60
CPX_MIPEMPHASIS_BALANCED (cplexConstants), 60
CPX_MIPEMPHASIS_BESTBOUND
INDEX

(cplexConstants), 60
CPX_MIPEMPHASIS_FEASIBILITY (cplexConstants), 60
CPX_MIPEMPHASIS_HIDDENFEAS (cplexConstants), 60
CPX_MIPEMPHASIS_OPTIMALITY (cplexConstants), 60
CPX_MKAPPA_AUTO (cplexConstants), 60
CPX_MKAPPA_FULL (cplexConstants), 60
CPX_MKAPPA_OFF (cplexConstants), 60
CPX_MKAPPA_SAMPLE (cplexConstants), 60
CPX_MIPORDER_BOUNDS (cplexConstants), 60
CPX_MIPORDER_COST (cplexConstants), 60
CPX_MIPORDER_SCALED_COST (cplexConstants), 60
CPX_MIPSEARCH_AUTO (cplexConstants), 60
CPX_MIPSEARCH_DYNAMIC (cplexConstants), 60
CPX_MIPSEARCH_TRADITIONAL (cplexConstants), 60
CPX_MIPSTART_AUTO (cplexConstants), 60
CPX_MIPSTART_CHECKFEAS (cplexConstants), 60
CPX_MIPSTART_REPAIR (cplexConstants), 60
CPX_MIPSTART_SOLVED_CUSTOM (cplexConstants), 60
CPX_MIPSTART_SOLVEMIP (cplexConstants), 60
CPX_NO_SOLN (cplexConstants), 60
CPX_NODESEL_BESTBOUND (cplexConstants), 60
CPX_NODESEL_BESTTEST (cplexConstants), 60
CPX_NODESEL_BESTTEST_ALT (cplexConstants), 60
CPX_NODESEL_DFS (cplexConstants), 60
CPX_NONBASIC_SOLN (cplexConstants), 60
CPX_OBJ_GAP (cplexConstants), 60
CPX_OFF (cplexConstants), 60
CPX_ON (cplexConstants), 60
CPX_PARALLEL_AUTO (cplexConstants), 60
CPX_PARALLEL_DETERMINISTIC (cplexConstants), 60
CPX_PARALLEL_OPPORTUNISTIC (cplexConstants), 60
CPX_PARAM_ADVIND (cplexConstants), 60
CPX_PARAM_AGGRCutLim (cplexConstants), 60
CPX_PARAM_AGGRFill (cplexConstants), 60
CPX_PARAM_AGGRIND (cplexConstants), 60
CPX_PARAM_ALL_MAX (cplexConstants), 60
CPX_PARAM_ALL_MIN (cplexConstants), 60
CPX_PARAM_APIENCODING (cplexConstants), 60
CPX_PARAM_AUXROOTTHREADS (cplexConstants), 60
CPX_PARAM_BARALGR (cplexConstants), 60
CPX_PARAM_BARCOLNZ (cplexConstants), 60
CPX_PARAM_BARCROSSALG (cplexConstants), 60
CPX_PARAM_BARDISPLAY (cplexConstants), 60
CPX_PARAM_BARDSTART (cplexConstants), 60
CPX_PARAM_BAREPCOMP (cplexConstants), 60
CPX_PARAM_BARGROWTH (cplexConstants), 60
CPX_PARAM_BARITLIM (cplexConstants), 60
CPX_PARAM_BARMAXCOR (cplexConstants), 60
CPX_PARAM_BAROBJRNG (cplexConstants), 60
CPX_PARAM_BARORDER (cplexConstants), 60
CPX_PARAM_BARPSTART (cplexConstants), 60
CPX_PARAM_BARQCPEP (cplexConstants), 60
CPX_PARAM_BARSTARTALG (cplexConstants), 60
CPX_PARAM_BASEINTERVAL (cplexConstants), 60
CPX_PARAM_BBIINTERVAL (cplexConstants), 60
CPX_PARAM_BNDSTRENIND (cplexConstants), 60
CPX_PARAM_BRDIR (cplexConstants), 60
CPX_PARAM_BTOL (cplexConstants), 60
CPX_PARAM_CALCQCSPDUALS (cplexConstants), 60
CPX_PARAM_CFACMUL (cplexConstants), 60
CPX_PARAM_CLIQUES (cplexConstants), 60
CPX_PARAM_CLOCKTYPE (cplexConstants), 60
CPX_PARAM_CLONELOG (cplexConstants), 60
CPX_PARAM_COEREDIND (cplexConstants), 60
CPX_PARAM_COLREADLIM (cplexConstants), 60
CPX_PARAM_CONFLICTDISPLAY (cplexConstants), 60
CPX_PARAM_COVERS (cplexConstants), 60
CPX_PARAM_CRAIND (cplexConstants), 60
CPX_PARAM_CUTLO (cplexConstants), 60
CPX_PARAM_CUTPASS (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
CPX_PARAM_CUTSFACCTOR (cplexConstants), 60
INDEX

60 CPX_PARAM_CUTUP (cplexConstants), 60
60 CPX_PARAM_DATACHECK (cplexConstants), 60
60 CPX_PARAM_DEPIND (cplexConstants), 60
60 CPX_PARAM_DETITLIM (cplexConstants), 60
60 CPX_PARAM_DISJ CUTS (cplexConstants), 60
60 CPX_PARAM_DIVETYPE (cplexConstants), 60
60 CPX_PARAM_DPRIIND (cplexConstants), 60
60 CPX_PARAM_EACHCUTLIM (cplexConstants), 60
60 CPX_PARAM_EACHCUTLIM_H (cplexConstants), 60
60 CPX_PARAM_EPAGAP (cplexConstants), 60
60 CPX_PARAM_EPGAP (cplexConstants), 60
60 CPX_PARAM_EPINT (cplexConstants), 60
60 CPX_PARAM_EPRHS (cplexConstants), 60
60 CPX_PARAM_EPRHS_H (cplexConstants), 60
60 CPX_PARAM_FASTMIP (cplexConstants), 60
60 CPX_PARAM_FEASOPTMODE (cplexConstants), 60
60 CPX_PARAM_FILEENCODING (cplexConstants), 60
60 CPX_PARAM_FLOWCOVERS (cplexConstants), 60
60 CPX_PARAM_FLOWPATHS (cplexConstants), 60
60 CPX_PARAM_FPHEUR (cplexConstants), 60
60 CPX_PARAM_FRACCAND (cplexConstants), 60
60 CPX_PARAM_FRACCUTS (cplexConstants), 60
60 CPX_PARAM_FRACPASS (cplexConstants), 60
60 CPX_PARAM_GUBCOVERS (cplexConstants), 60
60 CPX_PARAM_HUEFREQ (cplexConstants), 60
60 CPX_PARAM_IMPLBD (cplexConstants), 60
60 CPX_PARAM_INTSOLFILEPREFIX (cplexConstants), 60
60 CPX_PARAM_INTSOLFILEPREFIX_H (cplexConstants), 60
60 CPX_PARAM_INTSOLLIM (cplexConstants), 60
60 CPX_PARAM_INTSOLLIM_H (cplexConstants), 60
60 CPX_PARAM_ITLIM (cplexConstants), 60
60 CPX_PARAM_LANDPCUTS (cplexConstants), 60
60 CPX_PARAM_LBHEUR (cplexConstants), 60
60 CPX_PARAM_LPMETH (cplexConstants), 60
60 CPX_PARAM_MCFCUTS (cplexConstants), 60
60 CPX_PARAM_MEMORYEMPHASIS (cplexConstants), 60
60 CPX_PARAM_MIPBREDLP (cplexConstants), 60
60 CPX_PARAM_MIPDISPLAY (cplexConstants), 60
60 CPX_PARAM_MIPEMPHASIS (cplexConstants), 60
60 CPX_PARAM_MIPINTERVAL (cplexConstants), 60
60 CPX_PARAM_MIPORDIND (cplexConstants), 60
60 CPX_PARAM_MIPORDTYPE (cplexConstants), 60
60 CPX_PARAM_MIPSEARCH (cplexConstants), 60
60 CPX_PARAM_MIQPSTRAT (cplexConstants), 60
60 CPX_PARAM_MIRCUTS (cplexConstants), 60
60 CPX_PARAM_MPSLONGNUM (cplexConstants), 60
60 CPX_PARAM_NETDISPLAY (cplexConstants), 60
60 CPX_PARAM_NETEOPT (cplexConstants), 60
60 CPX_PARAM_NETPRHS (cplexConstants), 60
60 CPX_PARAM_NETPRHS_H (cplexConstants), 60
60 CPX_PARAM_NETTILIM (cplexConstants), 60
60 CPX_PARAM_NETPPRIIND (cplexConstants), 60
60 CPX_PARAM_NODEFILEIND (cplexConstants), 60
60 CPX_PARAM_NODELIM (cplexConstants), 60
60 CPX_PARAM_NODESEL (cplexConstants), 60
60 CPX_PARAM_NUMERICALEMPHASIS (cplexConstants), 60
60 CPX_PARAM_NZREADLIM (cplexConstants), 60
60 CPX_PARAM_OBJDIF (cplexConstants), 60
60 CPX_PARAM_OBLLIM (cplexConstants), 60
60 CPX_PARAM_OBLLIM (cplexConstants), 60
60 CPX_PARAM_PARALLELMODE (cplexConstants), 60
60 CPX_PARAM_PARAMDISPLAY (cplexConstants), 60
60 CPX_PARAM_PAREIND (cplexConstants), 60
60 CPX_PARAM_PERLIM (cplexConstants), 60
60 CPX_PARAM_POLISHAFTERDETTIME (cplexConstants), 60
60 CPX_PARAM_POLISHAFTEREPAGAP (cplexConstants), 60
60 CPX_PARAM_POLISHAFTEREPAGAP (cplexConstants), 60
60 CPX_PARAM_POLISHAFTERINTSOL

(cplexConstants), 60
CPX_PARAM_POLISHAFTERNODE (cplexConstants), 60
CPX_PARAM_POLISHAFTERTIME (cplexConstants), 60
CPX_PARAM_POLISHTIME (cplexConstants), 60
CPX_PARAM_POPULATELIM (cplexConstants), 60
CPX_PARAM_PPRIIND (cplexConstants), 60
CPX_PARAM_PREDUAL (cplexConstants), 60
CPX_PARAM_PREIND (cplexConstants), 60
CPX_PARAM_PRELINEAR (cplexConstants), 60
CPX_PARAM_PREPASS (cplexConstants), 60
CPX_PARAM_PRESLVND (cplexConstants), 60
CPX_PARAM_PRICELIM (cplexConstants), 60
CPX_PARAM_PROBE (cplexConstants), 60
CPX_PARAM_PROBEDETTIME (cplexConstants), 60
CPX_PARAM_PROBELTIME (cplexConstants), 60
CPX_PARAM_QMaketPSDIND (cplexConstants), 60
CPX_PARAM_QPMETHOD (cplexConstants), 60
CPX_PARAM_QPNZREADLIM (cplexConstants), 60
CPX_PARAM_RAMPUPDETTILIM (cplexConstants), 60
CPX_PARAM_RAMPUPDURATION (cplexConstants), 60
CPX_PARAM_RAMPUTILIM (cplexConstants), 60
CPX_PARAMRANDOMSEED (cplexConstants), 60
CPX_PARAM_REDUCE (cplexConstants), 60
CPX_PARAM_REINV (cplexConstants), 60
CPX_PARAM_RELAXPREIND (cplexConstants), 60
CPX_PARAM_RELOBJIDF (cplexConstants), 60
CPX_PARAM_REPAIRTRIES (cplexConstants), 60
CPX_PARAM_REPEATPRESOLVE (cplexConstants), 60
CPX_PARAM_REVERSEIND (cplexConstants), 60
CPX_PARAM_RFILEMUL (cplexConstants), 60
CPX_PARAM_RINSHEUR (cplexConstants), 60
CPX_PARAM_ROWREADLIM (cplexConstants), 60
CPX_PARAM_SCAIND (cplexConstants), 60
CPX_PARAM_SCRIND (cplexConstants), 60
CPX_PARAM_SFTALG (cplexConstants), 60
CPX_PARAM_SFTDISPLAY (cplexConstants), 60
CPX_PARAM_SFTITLIM (cplexConstants), 60
CPX_PARAM_SIMDISPLAY (cplexConstants), 60
CPX_PARAM_SINGLIM (cplexConstants), 60
CPX_PARAM_SINGTOL (cplexConstants), 60
CPX_PARAM_SOLNPOOLAGAP (cplexConstants), 60
CPX_PARAM_SOLNPOOLCAPACITY (cplexConstants), 60
CPX_PARAM_SOLNPOOLGAP (cplexConstants), 60
CPX_PARAM_SOLNPOOLINTENSITY (cplexConstants), 60
CPX_PARAM_SOLNPOOLREPLACE (cplexConstants), 60
CPX_PARAM_SOLUTIONTARGET (cplexConstants), 60
CPX_PARAM_STARTALG (cplexConstants), 60
CPX_PARAM_STRONGCANDLIM (cplexConstants), 60
CPX_PARAM_STRONGITLIM (cplexConstants), 60
CPX_PARAM_SUBALG (cplexConstants), 60
CPX_PARAM_SUBMIPNODELIM (cplexConstants), 60
CPX_PARAM_SYMMETRY (cplexConstants), 60
CPX_PARAM_THREADS (cplexConstants), 60
CPX_PARAM_TILIM (cplexConstants), 60
CPX_PARAM_TRELIM (cplexConstants), 60
CPX_PARAM_TUNINGDETTILIM (cplexConstants), 60
CPX_PARAM_TUNINGDISPLAY (cplexConstants), 60
CPX_PARAM_TUNINGMEASURE (cplexConstants), 60
CPX_PARAM_TUNINGREPEAT (cplexConstants), 60
CPX_PARAM_TUNINGTILIM (cplexConstants), 60
CPX_PARAM_VARSEL (cplexConstants), 60
CPX_PARAM_WORKDIR (cplexConstants), 60
CPX_PARAM_WORKMEM (cplexConstants), 60
CPX_PARAM_WRITELEVEL (cplexConstants), 60
INDEX

60

CPX_PARAM_XXXIND (cplexConstants), 60
CPX_PARAM_ZEROHALFCUTS (cplexConstants), 60
CPX_PARAMTYPE_DOUBLE (cplexConstants), 60
CPX_PARAMTYPE_INT (cplexConstants), 60
CPX_PARAMTYPE_LONG (cplexConstants), 60
CPX_PARAMTYPE_NONE (cplexConstants), 60
CPX_PARAMTYPE_STRING (cplexConstants), 60
CPX_PPRIIND_AUTO (cplexConstants), 60
CPX_PPRIIND_DEVEX (cplexConstants), 60
CPX_PPRIIND_FULL (cplexConstants), 60
CPX_PPRIIND_PARTIAL (cplexConstants), 60
CPX_PPRIIND_STEEP (cplexConstants), 60
CPX_PPRIIND_STEEPQSTART (cplexConstants), 60
CPX_PRECOL_AGG (cplexConstants), 60
CPX_PRECOL_FIX (cplexConstants), 60
CPX_PRECOL_LOW (cplexConstants), 60
CPX_PRECOL_OTHER (cplexConstants), 60
CPX_PRECOL_UP (cplexConstants), 60
CPX_PREREDUCE_DUALONLY (cplexConstants), 60
CPX_PREREDUCE_NOPRIMALORDUAL (cplexConstants), 60
CPX_PREREDUCE_PRIMALANDDUAL (cplexConstants), 60
CPX_PREREDUCE_PRIMALONLY (cplexConstants), 60
CPX_PREROW_AGG (cplexConstants), 60
CPX_PREROW_OTHER (cplexConstants), 60
CPX_PREROW_RED (cplexConstants), 60
CPX_PRIMAL_OBJC (cplexConstants), 60
CPX_PRIMAL_SOLN objc (cplexConstants), 60
CPX_SEMICONT (cplexConstants), 60
CPX_SEMIINT (cplexConstants), 60
CPX_SOLNPOOL_DIV (cplexConstants), 60
CPX_SOLNPOOL_FIFO (cplexConstants), 60
CPX_SOLNPOOL_FILTER_DIVERSITY (cplexConstants), 60
CPX_SOLNPOOL_FILTER_RANGE (cplexConstants), 60
CPX_SOLNPOOL_OBJ objc (cplexConstants), 60
CPX_SOLUTIONTARGET_AUTO (cplexConstants), 60
CPX_SOLUTIONTARGET_FIRSTORDER (cplexConstants), 60
CPX.solutiontarget_optimal_convex (cplexConstants), 60
CPX.solutiontarget_optimal_global (cplexConstants), 60
CPX_STAT_ABORT_DET_TIME_LIM (cplexConstants), 60
CPX_STAT_ABORT_DUAL_OBJ_LIM (cplexConstants), 60
CPX_STAT_ABORT_IT_LIM (cplexConstants), 60
CPX_STAT_ABORT_OBJ_LIM (cplexConstants), 60
CPX_STAT_ABORT_PRIM_OBJ_LIM (cplexConstants), 60
CPX_STAT_ABORT_TIME_LIM (cplexConstants), 60
CPX_STAT_ABORT_USER (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_CONTRADICTION (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_DET_TIME_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_IT_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_MEM_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_NODE_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_OBJ_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_TIME_LIM (cplexConstants), 60
CPX_STAT_CONFLICT_ABORT_USER (cplexConstants), 60
CPX_STAT_CONFLICT_FEASIBLE (cplexConstants), 60
CPX_STAT_CONFLICT_MINIMAL (cplexConstants), 60
CPX_STAT_FEASIBLE_RELAXED_INF (cplexConstants), 60
CPX_STAT_FEASIBLE_RELAXED_QUAD (cplexConstants), 60
CPX_STAT_FEASIBLE_RELAXED_SUM (cplexConstants), 60
CPX_STAT_FIRSTORDER (cplexConstants), 60
CPX_STAT_INFEASIBLE (cplexConstants), 60
CPX_STAT_INForUNBD (cplexConstants), 60

(continued)
INDEX

CPX_STAT_NUM_BEST (cplexConstants), 60
CPX_STAT_OPTIMAL (cplexConstants), 60
CPX_STAT_OPTIMAL_FACE_UNBOUNDED (cplexConstants), 60
CPX_STAT_OPTIMAL_INFEAS (cplexConstants), 60
CPX_STAT_OPTIMAL_RELAXED_INF (cplexConstants), 60
CPX_STAT_OPTIMAL_RELAXED_QAD (cplexConstants), 60
CPX_STAT_OPTIMAL_RELAXED_SUM (cplexConstants), 60
CPX_STAT_UNBOUNDED (cplexConstants), 60
CPX_STR_PARAM_MAX (cplexConstants), 60
CPX_SUM_COMP_SLACK (cplexConstants), 60
CPX_SUM_DUAL_INFEAS (cplexConstants), 60
CPX_SUM_DUAL_RESIDUAL (cplexConstants), 60
CPX_SUM_INDSLACK_INFEAS (cplexConstants), 60
CPX_SUM_INT_INFEAS (cplexConstants), 60
CPX_SUM_PI (cplexConstants), 60
CPX_SUM_PRIMAL_INFEAS (cplexConstants), 60
CPX_SUM_PRIMAL_RESIDUAL (cplexConstants), 60
CPX_SUM_PWLSLACK_INFEAS (cplexConstants), 60
CPX_SUM_QCPRIMAL_RESIDUAL (cplexConstants), 60
CPX_SUM_QCSLACK (cplexConstants), 60
CPX_SUM_QCSLACK_INFEAS (cplexConstants), 60
CPX_SUM_RED_COST (cplexConstants), 60
CPX_SUM_SCALED_DUAL_INFEAS (cplexConstants), 60
CPX_SUM_SCALED_DUAL_RESIDUAL (cplexConstants), 60
CPX_SUM_SCALED_PI (cplexConstants), 60
CPX_SUM_SCALED_PRIMAL_INFEAS (cplexConstants), 60
CPX_SUM_SCALED_PRIMAL_RESIDUAL (cplexConstants), 60
CPX_SUM_SCALED_RED_COST (cplexConstants), 60
CPX_SUM_SCALED_SLACK (cplexConstants), 60
CPX_SUM_SCALED_X (cplexConstants), 60
CPX_SUM_SLACK (cplexConstants), 60
CPX_SUM_X (cplexConstants), 60
CPX_TUNE_ABORT (cplexConstants), 60
CPX_TUNE_AVERAGE (cplexConstants), 60
CPX_TUNE_DETILIM (cplexConstants), 60
CPX_TUNE_MINMAX (cplexConstants), 60
CPX_TUNE_TILIM (cplexConstants), 60
CPX_TYPE_ANY (cplexConstants), 60
CPX_TYPE_SOS1 (cplexConstants), 60
CPX_TYPE_SOS2 (cplexConstants), 60
CPX_TYPE_USER (cplexConstants), 60
CPX_USECUT_FILTER (cplexConstants), 60
CPX_USECUT_FORCE (cplexConstants), 60
CPX_USECUT_PURGE (cplexConstants), 60
CPX_VARSEL_DEFAULT (cplexConstants), 60
CPX_VARSEL_MAXINFEAS (cplexConstants), 60
CPX_VARSEL_MININFEAS (cplexConstants), 60
CPX_VARSEL_PSEUDO (cplexConstants), 60
CPX_VARSEL_PSEUDOREDUCED (cplexConstants), 60
CPX_VARSEL_STRONG (cplexConstants), 60
CPX_WRITELEVEL_ALLVARS (cplexConstants), 60
CPX_WRITELEVEL_AUTO (cplexConstants), 60
CPX_WRITELEVEL_DISCRETEVARS (cplexConstants), 60
CPX_WRITELEVEL_NONZEROVARS (cplexConstants), 60
CPX_WRITELEVEL_NONZEROVARS (cplexConstants), 60
CPXaddcols (addColsCplex), 7
CPXaddfpdest (addFpDestCplex), 9
CPXaddindconst (addIndConstCplex), 10
CPXaddmipstarts (addMipstartsCplex), 11
CPXaddqconstr (addQConstrCplex), 12
CPXaddrows (addRowsCplex), 13
CPXbaropt (baroptCplex), 14
CPXbasicpresolve (basicPresolveCplex), 16
CPXboundsa (boundSaCplex), 17
CPXcheckaddcols (checkAddColsCplex), 18
CPXcheckaddrows (checkAddRowsCplex), 19
CPXcheckchgcoeftlist (checkChgCoefListCplex), 20
CPXcheckcopypctype
CPXMIP_DETTIME_LIM_FEAS (cplexConstants), 60
CPXMIP_DETTIME_LIM_INFEAS (cplexConstants), 60
CPXMIP_FAIL_FEAS (cplexConstants), 60
CPXMIP_FAIL_INFEAS (cplexConstants), 60
CPXMIP_FAIL_FEAS_NO_TREE (cplexConstants), 60
CPXMIP_FAIL_INFEAS_NO_TREE (cplexConstants), 60
CPXMIP_FEASIBLE (cplexConstants), 60
CPXMIP_FEASIBLE_RELAXED_INF (cplexConstants), 60
CPXMIP_FEASIBLE_RELAXED_QUAD (cplexConstants), 60
CPXMIP_FEASIBLE_RELAXED_SUM (cplexConstants), 60
CPXMIP_INFEASIBLE (cplexConstants), 60
CPXMIP_INForUNBD (cplexConstants), 60
CPXMIP_MEM_LIM_FEAS (cplexConstants), 60
CPXMIP_MEM_LIM_INFEAS (cplexConstants), 60
CPXMIP_NODE_LIM_FEAS (cplexConstants), 60
CPXMIP_NODE_LIM_INFEAS (cplexConstants), 60
CPXMIP_OPTIMAL (cplexConstants), 60
CPXMIP_OPTIMAL_INFEAS (cplexConstants), 60
CPXMIP_OPTIMAL_POPULATED (cplexConstants), 60
CPXMIP_OPTIMAL_POPULATED_TOL (cplexConstants), 60
CPXMIP_OPTIMAL_RELAXED_INF (cplexConstants), 60
CPXMIP_OPTIMAL_RELAXED_QUAD (cplexConstants), 60
CPXMIP_OPTIMAL_RELAXED_SUM (cplexConstants), 60
CPXMIP_OPTIMAL_TOL (cplexConstants), 60
CPXMIP_POPULATESOL_LIM (cplexConstants), 60
CPXMIP_SOL_LIM (cplexConstants), 60
CPXMIP_TIME_LIM_FEAS (cplexConstants), 60
CPXMIP_TIME_LIM_INFEAS (cplexConstants), 60
CPXMIP_UNBOUNDED (cplexConstants), 60
CPXmipopt (mipoptCPLEX), 189
CPXNET_NO_DISPLAY_OBJECTIVE (cplexConstants), 60
CPXNET_PENALIZED_OBJECTIVE (cplexConstants), 60
CPXNET_PRICE_AUTO (cplexConstants), 60
CPXNET_PRICE_MULT_PART (cplexConstants), 60
CPXNET_PRICEartial (cplexConstants), 60
CPXNET_TRUE_SORT_MULT_PART (cplexConstants), 60
CPXNetsum (newColsCPLEX), 190
CPXnewrows (newRowsCPLEX), 191
CPXobj (objSaCPLEX), 192
CPXopenCPLEX (openEnvCPLEX), 193
CPXordwrite (ordWriteCPLEX), 196
CPXPARAM_Advance (cplexConstants), 60
CPXPARAM_Barrier_Algorithm (cplexConstants), 60
CPXPARAM_Barrier_ColNonzeros (cplexConstants), 60
CPXPARAM_Barrier_ConvergeTol (cplexConstants), 60
CPXPARAM_Barrier_Crossover (cplexConstants), 60
CPXPARAM_Barrier_Ordering (cplexConstants), 60
CPXPARAM_Barrier_QCPConvergeTol (cplexConstants), 60
CPXPARAM_Barrier_StartAlg (cplexConstants), 60
CPXPARAM_ClockType (cplexConstants), 60
CPXPARAM_Conflict_Display (cplexConstants), 60
CPXPARAM_Conflict_Limits_Corrections (cplexConstants), 60
CPXPARAM_Conflict_Limits_Growth (cplexConstants), 60
CPXPARAM_Conflict_Limits_Iteration (cplexConstants), 60
CPXPARAM_Conflict_Limits_ObjRange (cplexConstants), 60
CPXPARAM_Conflict_Ordering (cplexConstants), 60
CPXPARAM_Conflict_QCPConvergeTol (cplexConstants), 60
CPXPARAM_Conflict_StartAlg (cplexConstants), 60
CPXPARAM_Price_Type (cplexConstants), 60
CPXPARAM_Price_Type_MULT_PART (cplexConstants), 60
CPXPARAM_Price_Type_PARTIAL (cplexConstants), 60
CPXPARAM_Price_Type_SORT_MULT_PART (cplexConstants), 60
CPXPARAM_Sum (CPXnewcols), 190
CPXPARAM_Time_LIM_FEAS (cplexConstants), 60
CPXPARAM_Time_LIM_INFEAS (cplexConstants), 60
CPXPARAM_DetTimeLimit (cplexConstants), 60
CPXPARAM_DistMIP_Rampup_DetTimeLimit (cplexConstants), 60
CPXPARAM_DistMIP_Rampup_Duration (cplexConstants), 60
CPXPARAM_DistMIP_Rampup_TimeLimit (cplexConstants), 60
CPXPARAM_Emphasis_Memory (cplexConstants), 60
CPXPARAM_Emphasis_MIP (cplexConstants), 60
CPXPARAM_Emphasis_Numerical (cplexConstants), 60
CPXPARAM_Feasopt_Mode (cplexConstants), 60
CPXPARAM_Feasopt_Tolerance (cplexConstants), 60
CPXPARAM_LPMeth (cplexConstants), 60
CPXPARAM_MIP_Cuts_Cliques (cplexConstants), 60
CPXPARAM_MIP_Cuts_Covers (cplexConstants), 60
CPXPARAM_MIP_Cuts_Disjunctive (cplexConstants), 60
CPXPARAM_MIP_Cuts_FlowCovers (cplexConstants), 60
CPXPARAM_MIP_Cuts_Gomory (cplexConstants), 60
CPXPARAM_MIP_Cuts_GUBCovers (cplexConstants), 60
CPXPARAM_MIP_Cuts_Implied (cplexConstants), 60
CPXPARAM_MIP_Cuts_LiftProj (cplexConstants), 60
CPXPARAM_MIP_Cuts_MCFCut (cplexConstants), 60
CPXPARAM_MIP_Cuts_MIRCut (cplexConstants), 60
CPXPARAM_MIP_Cuts_PathCut (cplexConstants), 60
CPXPARAM_MIP_Cuts_ZeroHalfCut (cplexConstants), 60
CPXPARAM_MIP_Display (cplexConstants), 60
CPXPARAM_MIP_Interval (cplexConstants), 60
CPXPARAM_MIP_Limits_AggForCut (cplexConstants), 60
CPXPARAM_MIP_Limits_AuxRootThreads (cplexConstants), 60
CPXPARAM_MIP_Limits_CutPasses (cplexConstants), 60
CPXPARAM_MIP_Limits_CutsFactor (cplexConstants), 60
CPXPARAM_MIP_Limits_EachCutLimit (cplexConstants), 60
CPXPARAM_MIP_Limits_GomoryCand (cplexConstants), 60
CPXPARAM_MIP_Limits_GomoryPass (cplexConstants), 60
CPXPARAM_MIP_Limits_Nodes (cplexConstants), 60
CPXPARAM_MIP_Limits_PolishTime (cplexConstants), 60
CPXPARAM_MIP_Limits_Populate (cplexConstants), 60
CPXPARAM_MIP_Limits_ProbeDetTime (cplexConstants), 60
CPXPARAM_MIP_Limits_ProbeTime (cplexConstants), 60
CPXPARAM_MIP_Limits_RepairTries (cplexConstants), 60
CPXPARAM_MIP_Limits_Solutions (cplexConstants), 60
CPXPARAM_MIP_Limits_StrongCand (cplexConstants), 60
CPXPARAM_MIP_Limits_StrongIt (cplexConstants), 60
CPXPARAM_MIP_Limits_SubMIPNodeLim (cplexConstants), 60
CPXPARAM_MIP_Limits_TreeMemory (cplexConstants), 60
CPXPARAM_MIP_OrderType (cplexConstants), 60
CPXPARAM_MIP_PolishAfter_AbsMIPGap (cplexConstants), 60
CPXPARAM_MIP_PolishAfter_DetTime (cplexConstants), 60
CPXPARAM_MIP_PolishAfter_MIPGap (cplexConstants), 60
CPXPARAM_MIP_PolishAfter_Solutions (cplexConstants), 60
CPXPARAM_MIP_PolishAfter_Time (cplexConstants), 60
INDEX

(cplexConstants), 60
CPXPARAM_MIP_Pool_AbsGap
(cplexConstants), 60
CPXPARAM_MIP_Pool_Capacity
(cplexConstants), 60
CPXPARAM_MIP_Pool_Intensity
(cplexConstants), 60
CPXPARAM_MIP_Pool_RelGap
(cplexConstants), 60
CPXPARAM_MIP_Pool.Replace
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Backtrack
(cplexConstants), 60
CPXPARAM_MIP_Strategy_BBInterval
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Branch
(cplexConstants), 60
CPXPARAM_MIP_Strategy_CallbackReducedLP
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Dive
(cplexConstants), 60
CPXPARAM_MIP_Strategy_File
(cplexConstants), 60
CPXPARAM_MIP_Strategy_FPHeur
(cplexConstants), 60
CPXPARAM_MIP_Strategy_HeuristicFreq
(cplexConstants), 60
CPXPARAM_MIP_Strategy_KappaStats
(cplexConstants), 60
CPXPARAM_MIP_Strategy_LBHeur
(cplexConstants), 60
CPXPARAM_MIP_Strategy_MIQCPStrat
(cplexConstants), 60
CPXPARAM_MIP_Strategy_NodeSelect
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Order
(cplexConstants), 60
CPXPARAM_MIP_Strategy_PresolveNode
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Probe
(cplexConstants), 60
CPXPARAM_MIP_Strategy_RINSHeur
(cplexConstants), 60
CPXPARAM_MIP_Strategy_Search
(cplexConstants), 60
CPXPARAM_MIP_Strategy_StartAlgorithm
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubAlgorithm
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubMIPScale
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubMIPStartAlg
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubMIPSubAlg
(cplexConstants), 60
CPXPARAM_MIP_Strategy_VariableSelect
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_AbsMIPGap
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_Integrality
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_LowerCutoff
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_MIPGap
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_ObjDifference
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_RelObjDifference
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_UpperCutoff
(cplexConstants), 60
CPXPARAM_MultiObjective_Display
(cplexConstants), 60
CPXPARAM_Network_Display
(cplexConstants), 60
CPXPARAM_Network_Iterations
(cplexConstants), 60
CPXPARAM_Network_NetFind
(cplexConstants), 60
CPXPARAM_Network_Pricing
(cplexConstants), 60
CPXPARAM_Network_Tolerances_Feasibility
(cplexConstants), 60
CPXPARAM_Network_Tolerances_Optimality
(cplexConstants), 60
CPXPARAM_Output_CloneLog
(cplexConstants), 60
CPXPARAM_Output_IntSolFilePrefix
(cplexConstants), 60
CPXPARAM_Output_MPSLong
(cplexConstants), 60
CPXPARAM_Output_WriteLevel
(cplexConstants), 60
CPXPARAM_Parallel
(cplexConstants), 60
CPXPARAM_ParamDisplay
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubMIPStartAlg
(cplexConstants), 60
CPXPARAM_MIP_Strategy_SubMIPSubAlg
(cplexConstants), 60
CPXPARAM_MIP_Strategy_VariableSelect
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_AbsMIPGap
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_Integrality
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_LowerCutoff
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_MIPGap
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_ObjDifference
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_RelObjDifference
(cplexConstants), 60
CPXPARAM_MIP_Tolerances_UpperCutoff
(cplexConstants), 60
CPXPARAM_MultiObjective_Display
(cplexConstants), 60
CPXPARAM_Network_Display
(cplexConstants), 60
CPXPARAM_Network_Iterations
(cplexConstants), 60
CPXPARAM_Network_NetFind
(cplexConstants), 60
CPXPARAM_Network_Pricing
(cplexConstants), 60
CPXPARAM_Network_Tolerances_Feasibility
(cplexConstants), 60
CPXPARAM_Network_Tolerances_Optimality
(cplexConstants), 60
CPXPARAM_Output_CloneLog
(cplexConstants), 60
CPXPARAM_Output_IntSolFilePrefix
(cplexConstants), 60
CPXPARAM_Output_MPSLong
(cplexConstants), 60
CPXPARAM_Output_WriteLevel
(cplexConstants), 60
CPXPARAM_Parallel (cplexConstants), 60
CPXPARAM_ParamDisplay (cplexConstants), 60
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPXPARAM_Preprocessing_Aggregator</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_BoundStrength</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_CoeffReduce</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Dependency</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Dual</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Fill</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Folding</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Linear</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_NumPass</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Presolve</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_QCPDuals</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_QPMakePSD</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Reduce</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Relax</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_RepeatePresolve</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Preprocessing_Symmetry</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_QPMethod</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_RandomSeed</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_APIEncoding</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_Constraints</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_DataCheck</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_FileEncoding</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_Nonzeros</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_QPNonzeros</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_Scale</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Read_Variables</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Record</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_ScreenOutput</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Algorithm</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Display</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Sifting_Iterations</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Crash</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_DGradient</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Display</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_LimitsIterations</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_LowerObj</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_Perturbation</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_Singularity</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Limits_UpperObj</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Perturbation_Constant</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Perturbation_Indicator</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_PGradient</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Pricing</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Refactor</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Feasibility</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Markowitz</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_Simplex_Tolerances_Optimality</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_SolutionTarget</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_THREADS</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>CPXPARAM_TimeLimit</td>
<td>(cplexConstants), 60</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Page Numbers</td>
<td></td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------</td>
<td></td>
</tr>
<tr>
<td><code>feasOptCPLEX</code></td>
<td>98</td>
<td></td>
</tr>
<tr>
<td><code>fileputCPLEX</code></td>
<td>47, 99, 182, 194</td>
<td></td>
</tr>
<tr>
<td><code>flushChannelCPLEX</code></td>
<td>95, 100, 101, 105</td>
<td></td>
</tr>
<tr>
<td><code>flushStdChannelsCPLEX</code></td>
<td>95, 100, 101, 105</td>
<td></td>
</tr>
<tr>
<td><code>freePresolveCPLEX</code></td>
<td>102</td>
<td></td>
</tr>
<tr>
<td><code>getBaseCPLEX</code></td>
<td>103</td>
<td></td>
</tr>
<tr>
<td><code>getBestObjValCPLEX</code></td>
<td>104, 136</td>
<td></td>
</tr>
<tr>
<td><code>getChannelsCPLEX</code></td>
<td>95, 100, 101, 105</td>
<td></td>
</tr>
<tr>
<td><code>getColParmCPLEX</code></td>
<td>106, 157</td>
<td></td>
</tr>
<tr>
<td><code>getColIndexCPLEX</code></td>
<td>111</td>
<td></td>
</tr>
<tr>
<td><code>getColNameCPLEX</code></td>
<td>110</td>
<td></td>
</tr>
<tr>
<td><code>getColsCPLEX</code></td>
<td>111</td>
<td></td>
</tr>
<tr>
<td><code>getConflictCPLEX</code></td>
<td>113, 207, 209</td>
<td></td>
</tr>
<tr>
<td><code>getConflictExtCPLEX</code></td>
<td>114</td>
<td></td>
</tr>
<tr>
<td><code>getCutoffCPLEX</code></td>
<td>115</td>
<td></td>
</tr>
<tr>
<td><code>getDblParmCPLEX</code></td>
<td>116</td>
<td></td>
</tr>
<tr>
<td><code>getDblQualCPLEX</code></td>
<td>117</td>
<td></td>
</tr>
<tr>
<td><code>getDbsCntCPLEX</code></td>
<td>118</td>
<td></td>
</tr>
<tr>
<td><code>getDjCPLEX</code></td>
<td>119</td>
<td></td>
</tr>
<tr>
<td><code>getErrorStrCPLEX</code></td>
<td>120, 178</td>
<td></td>
</tr>
<tr>
<td><code>getGradCLEX</code></td>
<td>121</td>
<td></td>
</tr>
<tr>
<td><code>getIndConstrCLEX</code></td>
<td>122</td>
<td></td>
</tr>
<tr>
<td><code>getInfoDblParmCPLEX</code></td>
<td>123</td>
<td></td>
</tr>
<tr>
<td><code>getInfoIntParmCPLEX</code></td>
<td>124, 126</td>
<td></td>
</tr>
<tr>
<td><code>getInfoLongParmCPLEX</code></td>
<td>125</td>
<td></td>
</tr>
<tr>
<td><code>getInfoParmNameCPLEX</code></td>
<td>126</td>
<td></td>
</tr>
<tr>
<td><code>getInfoParmTypeCPLEX</code></td>
<td>127, 132</td>
<td></td>
</tr>
<tr>
<td><code>getInfoParmValCPLEX</code></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td><code>getItsCtplex</code></td>
<td>129</td>
<td></td>
</tr>
<tr>
<td><code>getLogFileCPLEX</code></td>
<td>130, 216</td>
<td></td>
</tr>
<tr>
<td><code>getLogFileNameCPLEX</code></td>
<td>131</td>
<td></td>
</tr>
<tr>
<td><code>getLowBndsIdsCPLEX</code></td>
<td>132</td>
<td></td>
</tr>
<tr>
<td><code>getLowerBndsCPLEX</code></td>
<td>28, 133, 134</td>
<td></td>
</tr>
<tr>
<td><code>getMethodCLEX</code></td>
<td>135</td>
<td></td>
</tr>
<tr>
<td><code>getMPIpGapCPLEX</code></td>
<td>136</td>
<td></td>
</tr>
<tr>
<td><code>getMIPstartIndexCPLEX</code></td>
<td>137</td>
<td></td>
</tr>
<tr>
<td><code>getMIPstartNameCPLEX</code></td>
<td>138</td>
<td></td>
</tr>
<tr>
<td><code>getMIPstartsCPLEX</code></td>
<td>139</td>
<td></td>
</tr>
<tr>
<td><code>getNumColsCPLEX</code></td>
<td>29, 140</td>
<td></td>
</tr>
<tr>
<td><code>getNumMIPstartsCPLEX</code></td>
<td>141</td>
<td></td>
</tr>
<tr>
<td><code>getNumNzCPLEX</code></td>
<td>142</td>
<td></td>
</tr>
<tr>
<td><code>getNumQConstrsCPLEX</code></td>
<td>143</td>
<td></td>
</tr>
<tr>
<td><code>getNumQpNZCPLEX</code></td>
<td>144</td>
<td></td>
</tr>
<tr>
<td><code>getNumQuadCPLEX</code></td>
<td>145</td>
<td></td>
</tr>
<tr>
<td><code>getNumRowsCPLEX</code></td>
<td>29, 146</td>
<td></td>
</tr>
<tr>
<td><code>getObjCLEX</code></td>
<td>147</td>
<td></td>
</tr>
<tr>
<td><code>getObjDirCLEX</code></td>
<td>148</td>
<td></td>
</tr>
<tr>
<td><code>getObjNameCLEX</code></td>
<td>149</td>
<td></td>
</tr>
<tr>
<td><code>getObjOffsetCLEX</code></td>
<td>150</td>
<td></td>
</tr>
<tr>
<td><code>getObjValCLEX</code></td>
<td>136, 151</td>
<td></td>
</tr>
<tr>
<td><code>getOrderCLEX</code></td>
<td>152</td>
<td></td>
</tr>
<tr>
<td><code>getParmHierNameCLEX</code></td>
<td>153</td>
<td></td>
</tr>
<tr>
<td><code>getParmNameCLEX</code></td>
<td>154</td>
<td></td>
</tr>
<tr>
<td><code>getParmNumCPLEX</code></td>
<td>155</td>
<td></td>
</tr>
<tr>
<td><code>getParmTypeCLEX</code></td>
<td>156</td>
<td></td>
</tr>
<tr>
<td><code>getParmValCLEX</code></td>
<td>82, 157</td>
<td></td>
</tr>
<tr>
<td><code>getPhase1CntCLEX</code></td>
<td>157</td>
<td></td>
</tr>
<tr>
<td><code>getPiCLEX</code></td>
<td>158</td>
<td></td>
</tr>
<tr>
<td><code>getPreStatCLEX</code></td>
<td>159</td>
<td></td>
</tr>
<tr>
<td><code>getProbNameCLEX</code></td>
<td>160</td>
<td></td>
</tr>
<tr>
<td><code>getProbTypeCLEX</code></td>
<td>38, 161, 225</td>
<td></td>
</tr>
<tr>
<td><code>getProbVarCPLEX</code></td>
<td>162</td>
<td></td>
</tr>
<tr>
<td><code>getQConstrCPLEX</code></td>
<td>163</td>
<td></td>
</tr>
<tr>
<td><code>getQPcoefsCPLEX</code></td>
<td>164</td>
<td></td>
</tr>
<tr>
<td><code>getQuadCLEX</code></td>
<td>165</td>
<td></td>
</tr>
<tr>
<td><code>getRedLpcLEX</code></td>
<td>166</td>
<td></td>
</tr>
<tr>
<td><code>getRhsCLEX</code></td>
<td>167</td>
<td></td>
</tr>
<tr>
<td><code>getRngValCLEX</code></td>
<td>168</td>
<td></td>
</tr>
<tr>
<td><code>getRowIndexCLEX</code></td>
<td>169</td>
<td></td>
</tr>
<tr>
<td><code>getRowConfCLEX</code></td>
<td>98, 170</td>
<td></td>
</tr>
<tr>
<td><code>getRowsCLEX</code></td>
<td>172</td>
<td></td>
</tr>
<tr>
<td><code>getSenseCLEX</code></td>
<td>173</td>
<td></td>
</tr>
<tr>
<td><code>getSiftCtplex</code></td>
<td>174</td>
<td></td>
</tr>
<tr>
<td><code>getSiftPresCplex</code></td>
<td>175</td>
<td></td>
</tr>
<tr>
<td><code>getSlackCLEX</code></td>
<td>176</td>
<td></td>
</tr>
<tr>
<td><code>getStatCLEX</code></td>
<td>15, 96, 98, 177, 186, 187, 189, 190, 199, 201, 226</td>
<td></td>
</tr>
<tr>
<td><code>getStatStrCLEX</code></td>
<td>120, 178, 226</td>
<td></td>
</tr>
<tr>
<td><code>getStrCplex</code></td>
<td>179</td>
<td></td>
</tr>
<tr>
<td><code>getSubMethodCLEX</code></td>
<td>180</td>
<td></td>
</tr>
<tr>
<td><code>getTimeCLEX</code></td>
<td>182</td>
<td></td>
</tr>
<tr>
<td><code>getUppBndsIdsCLEX</code></td>
<td>183</td>
<td></td>
</tr>
<tr>
<td><code>getUpperBndsCLEX</code></td>
<td>28, 183, 184</td>
<td></td>
</tr>
<tr>
<td><code>getVersionCLEX</code></td>
<td>185</td>
<td></td>
</tr>
<tr>
<td><code>hybbaroptCLEX</code></td>
<td>185</td>
<td></td>
</tr>
<tr>
<td><code>hybnetoptCLEX</code></td>
<td>186</td>
<td></td>
</tr>
</tbody>
</table>
INDEX


isCPLEXchanPointer (cplexPtr-class), 83
isCPLEXchanPointer, cplexPtr-method (cplexPtr-class), 83
isCPLEXenvPointer (cplexPtr-class), 83
isCPLEXenvPointer, cplexPtr-method (cplexPtr-class), 83
isCPLEXfilePointer (cplexPtr-class), 83
isCPLEXfilePointer, cplexPtr-method (cplexPtr-class), 83
isCPLEXprobPointer (cplexPtr-class), 83
isCPLEXprobPointer, cplexPtr-method (cplexPtr-class), 83
isCPLEXtermPointer (cplexPtr-class), 83
isCPLEXtermPointer, cplexPtr-method (cplexPtr-class), 83
isNULLpointerCPLEX (cplexPtr-class), 83
isNULLpointerCPLEX, cplexPtr-method (cplexPtr-class), 83

lpoptCPLEX, 188
mipoptCPLEX, 189
newColsCPLEX, 190
newRowsCPLEX, 191
objSaCPLEX, 192
openFileCPLEX, 9, 47, 85, 99, 182, 194, 216
openProbCPLEX, 48, 195
ordWriteCPLEX, 196

preslvWriteCPLEX, 197
presolveCPLEX, 198
primoptCPLEX, 199
printTerminateCPLEX, 44, 94, 200, 221
qpoptCPLEX, 200

readCopyBaseCPLEX, 201
readCopyMIPstartsCPLEX, 202
readCopyOrderCPLEX, 203
readCopyParmCPLEX, 204
readCopyProbCPLEX, 197, 205
readCopySolCPLEX, 206
refineConflictCPLEX, 207
refineConflictExtCPLEX, 208
refineMIPstartConflictCPLEX, 209
refineMIPstartConflictExtCPLEX, 210
return_codeCPLEX, 82, 211
rhsSaCPLEX, 212

setDbParmCPLEX, 213
setDefaultParmCPLEX, 214
setIntParmCPLEX, 215, 218
setLogFileCPLEX, 130, 216
setLogFileNameCPLEX, 217
setLongParmCPLEX, 218
setObjDirCPLEX, 219
setStrParmCPLEX, 220
setTerminateCPLEX, 44, 94, 200, 221
siftoptCPLEX, 222
solnInfoCPLEX, 15, 83, 96, 98, 186, 187, 189, 190, 199, 201, 223, 224
solutionCPLEX, 15, 83, 96, 98, 186, 187, 189, 190, 199, 201, 223, 224
solWriteCPLEX, 225
status_codeCPLEX, 82, 226
summary, cplexPtr-method (cplexPtr-class), 83
tightenBndsCPLEX, 32, 227
tuneParmCPLEX, 228

unscaleProbCPLEX, 229

writeMIPstartsCPLEX, 230
writeParmCPLEX, 231
writeProbCPLEX, 232