

# Package ‘hexfont’

February 14, 2023

**Type** Package

**Title** 'GNU Unifont' Hex Fonts

**Version** 0.3.1

**Description** Contains all the hex font files from the 'GNU Unifont Project' <<https://unifoundry.com/unifont/>> compressed by 'xz'. 'GNU Unifont' is a duospaced bitmap font that attempts to cover all the official Unicode glyphs plus several of the artificial scripts in the '(Under-)ConScript Unicode Registry' <<http://www.kreativekorp.com/ucsur/>>. Provides a convenience function for loading in several of them at the same time as a 'bittermelon' bitmap font object for easy rendering of the glyphs in an 'R' terminal or graphics device.

**URL** <https://github.com/trevorld/hexfont>

**BugReports** <https://github.com/trevorld/hexfont/issues>

**License** GPL (>= 2)

**Imports** bittermelon (>= 1.1.0), utils

**Suggests** knitr, oblicubes, rmarkdown, testthat, Unicode

**VignetteBuilder** knitr, rmarkdown

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**NeedsCompilation** no

**Author** Trevor L Davis [aut, cre] (<<https://orcid.org/0000-0001-6341-4639>>),  
GNU Unifont authors [cph]

**Maintainer** Trevor L Davis <trevor.l.davis@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-02-14 09:30:02 UTC

## R topics documented:

unifont . . . . .	2
unifont_combining . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

unifont

*Load Unifont font***Description**

The function `unifont()` loads in several GNU Unifont hex files as a single `bittermelon::bm_font()` object.

**Usage**

```
unifont(upper = TRUE, jp = FALSE, csur = TRUE, sample = FALSE, ucp = NULL)
```

**Arguments**

<code>upper</code>	Include glyphs above the Unicode Basic Multilingual plane.
<code>jp</code>	Use Japanese version of Chinese characters.
<code>csur</code>	Include (Under-)Conscript Unicode Registry glyphs.
<code>sample</code>	Add circle to "Combining" characters.
<code>ucp</code>	Character vector of Unicode Code Points: glyphs not in this vector won't be read in. If NULL (default) read every glyph in the font.

**Value**

A `bittermelon::bm_font()` object.

**Examples**

```
# Much faster to load only the subset of GNU Unifont one needs
# Mandarin Chinese
if (require("bittermelon")) {
  s <- "\uff32\u5f88\u68d2\uff01"
  font <- unifont(ucp = str2ucp(s))
  bm <- as_bm_bitmap(s, font = font)
  print(bm, px = px_ascii)
}

# Emoji
if (require("bittermelon")) {
  s <- "\U0001f42d\U0001f432\U0001f435"
  font <- unifont(ucp = str2ucp(s))
  bm <- as_bm_bitmap(s, font = font)
  print(bm, px = px_ascii)
}
```

---

unifont_combining	<i>Get combining character code points</i>
-------------------	--

---

**Description**

unifont\_combining() returns a character vector of the code points for all the "combining" characters in Unifont.

**Usage**

```
unifont_combining(upper = TRUE, csur = TRUE, unicode = FALSE)
```

**Arguments**

upper	Include glyphs above the Unicode Basic Multilingual plane.
csur	Include (Under-)Conscript Unicode Registry glyphs.
unicode	Include combining glyphs assigned by the Unicode Consortium (i.e. not ones in the Private Use Area like the CSUR ones). By default FALSE since <a href="#">bittermelon::bm_compose()</a> can usually guess that a Unicode Consortium assigned glyph is a combining glyph by using <a href="#">Unicode::u_char_property()</a> .

**Value**

A character vector of Unicode code points

**See Also**

Can be used with the pua\_combining argument of [bittermelon::bm\\_compose\(\)](#) and [bittermelon::as\\_bm\\_bitmap\(\)](#).

**Examples**

```
uc <- unifont_combining()
print(uc)

# Tengwar with combining glyphs
if (require("bittermelon")) {
  s <- "\ue004\ue014\ue04a\ue005\ue000\ue040\ue022\ue04a\ue003\ue04e"
  font <- unifont(ucp = str2ucp(s))
  bml <- as_bm_list(s, font = font)
  to_raise <- which(names(bml) %in% c("U+E04A", "U+E04E"))
  bml[to_raise] <- bm_shift(bml[to_raise], top = 1L)
  bml <- bm_compose(bml, pua_combining = uc)
  bml <- bm_pad(bml, type = "trim", left = 1L, right = 0L)
  bm <- bm_call(bml, cbind)
  print(bm, px = px_ascii)
}
```

# Index

bittermelon::as\_bm\_bitmap(), 3  
bittermelon::bm\_compose(), 3  
bittermelon::bm\_font(), 2  
  
Unicode::u\_char\_property(), 3  
unifont, 2  
unifont\_combining, 3