The NewComputerModern FontFamily

Antonis Tsolomitis • University of the Aegean • Department of Mathematics

The New Computer Modern FontFamily version 4.2

Antonis Tsolomitis

January 2, 2023

Contents

1 Introduction

The NewComputerModern FontFamily is a huge extension ("huge" in terms of the number of additional glyphs) of the `lm` fonts. It is not just a family adding random missing glyphs but it adds support for several more languages and shapes needed for academic (and not only) work. Currently it supports among others, Greek[1], Cyrillic[2], Hebrew, Cherokee and Coptic. Since it supports diacritics stacking the number of languages that use the Latin alphabet is greatly expanded. Diacritics stacking is also needed for Greek for papyrological work and this is also supported.

Version 4.0 adds to the classic design of computer modern new shapes for Latin and Greek, in particular it adds families for Medieval Latin and Uncial Greek matching in style to the main family.

In terms of weights and sizes, all of its shapes come in Regular, Book weights at 10 and 8 point sizes and in Bold at 10 points.

Mathematics is also supported in Regular and Book weights, currently providing a full coverage of the Unicode Math blocks (with a few more glyphs needed for Mathematics that Unicode has forgotten to encode).

What follows is a sequence of commands and results so as to show how to access all features of the fonts. Character tables are also included.

2 How to load the fonts

The simpler way to load the fonts is through the `fontsetup` package. The command

    `\usepackage[default]{fontsetup}`

will load the Book weight of the NewCM family, and

    `\usepackage[olddefault]{fontsetup}`

will load the Regular weight.

Also notice that the fonts support the microtype package for fine typographic tuning. See the documentation of microtype for this.

---

[1]from Claudio Beccari's Greek.

[2]from the `cmu` package.

3 The Latin alphabet

3.1 Ligatures and stylistic alternatives in Latin

The Serif font includes additional ligatures fb ffb ffh ffj ffk fft fh fj ft fk and the same with longs instead of f in the default liga table (in addition to the default fi fl ffi ffl ff). It also includes an alternative k (in the cv01 table) and sp ch ck ct st in the dlig table. Finally it also inludes "end" versions for the letters a, e, m, n and r in the cv02 table. To access the alternative k load the relative font (here the Book weight) with

`\setmainfont[CharacterVariant=1]{NewCM10-Book.otf}`

To load the same font with the dlig table enabled use

`\setmainfont[RawFeature=+dlig]{NewCM10-Book.otf}`

and to load the font with endings variations use

`\setmainfont[CharacterVariant=2]{NewCM10-Regular.otf}`

Of course the above can be mixed separating the optional arguments with comma, or one can define a custom font say by using

`\newfontfamily\myfont[<options to enable>]{NewCM10-Regular.otf}`

Regular	k	a e m n r	sp ch ck ct st
cv01	k		
cv02		a, e, m, n, r	
dlig			sp ch ck ct st

3.2 Old Italic

The fonts also fully support the Old Italic Unicode block (U10300–U1032F) in the Sans font. For example, the slots U10307, U10310, U10312, U10314, U1031F and U1032F are 𐌇𐌐𐌒𐌔𐌟𐌯.

3.3 Diacritics Stacking

Diacritics stacking is supported at least for Regular, Book and Bold upright seriffed fonts. In the margin you can see an example of stacking on the letter "x". If you need to enter these accents you can use the **\char** command or just copy-paste from the following line (from this pdf file or the provided source TEX file):

Some of the upper accents

ˋ ˊ ˆ ˜ ˗ ― ˘ ˙ ˙˙ ˀ ˚ ˮ

ˇ ˈ ˝ ˶ ⌃ ˒ ˓ ˟ ˌ ˥ ˦ ⌣ ˜ ⌢

Some of the lower accents

ˠ ˎ ˏ ˧ ˦ ˩˓ ˨ � ˧ ˗ ˙ ˙˙ ˳ ˌ

ˌ ˥ ˷ ˬ ˄ ˯ ˰ ˜ ˗ ― ˒ ˻ ˼ ˜ ˭ ˮ ˥ ↔

5
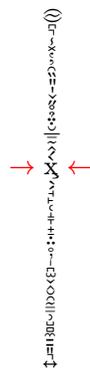
Please note that stacking is by default supported with xetex. With luatex you have to add the option `Renderer=Harfbuzz`, say by
`\addfontfeature{Renderer=Harfbuzz}`

Also notice that your text editor may not support stacking. The editor may show the accents one after the other, but the pdf produced by xetex or luatex will have the accents stacked.

4 Greek

The full Unicode Greek block is supported, which is

- U0370–U03FF for monotonic, where missing glyphs, such as Heta (Ⱶ), Pamphilian digamma (Ͷ) etc, have been added. For example, it is now possible to write

  βιβλίο instead of βιβλίο.

- U1F00–U1FFF for polytonic, and

- U10140–U1018F for ancient Greek numbers.

Θεώρημα 4.1 (Πυθαγόρειον) *Ἐν τοῖς ὀρθογωνίοις τριγώνοις τὸ ἀπὸ τῆς τὴν ὀρθὴν γωνίαν ὑποτεινούσης πλευρᾶς τετράγωνον ἴσον ἐστὶ τοῖς ἀπὸ τῶν τὴν ὀρθὴν γωνίαν περιεχουσῶν πλευρῶν τετραγώνοις.*

Small Caps is included (in Mono font too) and all polytonic accents of Greek. Ypogegrammeni is the default for all characters including Small Caps and prosgegrammeni is offered as an alternative shape in the `ss01` lookup table:

	ypogegrammeni	prosgegrammeni
regular	ᾌ ᾟ ᾯ ᾼῌῼ	ᾌ ᾟ ᾯ Αͺ,Ηͺ,Ωͺ
sans	ᾌ ᾟ ᾯ ᾼῌῼ	ᾌ ᾟ ᾯ Α,Η,Ω,
mono	ᾌ ᾟ ᾯ ᾼῌῼ	ᾌ,ᾟ,ᾯ, Α,Η,Ω,

The prosgegrammeni alternates can be accessed with

    \textprosgegrammeni{<text>}
or the
    {\prosgegrammeni <text>}

of the `fontsetup` package.

4.1 Other character variants

Guillemots (left and right) have a different shape for Greek. For this to work the fonts must be loaded with the cv04 character variant.

Compare the default guillemots: «» with Greek guillemots: «».

There is a serious problem with Unicode and the Greek anoteleia (U0387); the Greek semicolon. Unicode "thinks" that this character is the same with periodcentered (U00B7). This influences the way keyboards are configured by several vendors such as xorg. Anoteleia is a dot written at x-height and not at 1/2 the x-height as the periodcentered. Although Unicode recognizes the problem[3], althought they recognize that with their current standard you can not correctly write the Greek language, they refuse to fix it, justifying it by saying the magical words "backwards compatibility" (to a …mistake, one could add).

NewComputerModern can not allow this, as it defies the purpose of its existence, which is to properly write every supported language. So enabling the CharacterVariant 04 (cv04) in addition to correct guillemots for Greek it maps periodcentered (produced by the keyboards (in Greek Linux keyboards by AltGr+q) to proper anoteleia.

It also fixes a long standing issue with the Greek apostrophe (')(U1FBD) which is not the same with quoteright (')(U2019). U1FBD named as "Greek Koronis" by Unicode is the proper character.

Another problem that has to do with quotes inside quotes. The internal quotes in Greek should be written with the characters quotedblleft and quotedblbase[4]. For example, this is correct for Greek

<p style="text-align:center">«άλφα "βήτα„»</p>

But the keyboards only produce quotesingle which is already mapped to apostrophe and it is difficult to remember the names "quotedblleft" and "quotedblbase". So when enabling cv04 one can define the commands

    `\newcommand\leftgrquotes{\char"201C}`

and

    `\newcommand\rightgrquotes{\char"201E}`

for the rare case one needs quotes inside quotes. The `fontsetup` package does this automatically for Greek if the `xgreek` package has been loaded *before* the `fontsetup` package. Otherwise, for non-Greek documents with small passages of Greek, the author may enable `cv04` by creating a custom command such as

    `\newcommand\propergreek[CharacterVariant=4]{NewCM10-Book.otf}`

A phrase with Greek quotes inside quotes, proper anoteleia, and proper apostrophe is

<p style="text-align:center">«φώναζε: "απ᾽ έξω την προπαίδεια„»· σαν εκδίκηση ακουγόταν…</p>

4.2 Prosodic symbols

In Greek philology and in linguistics it is often needed to stack accent-type symbols above letters, even if they are not vowels. Although rare in writings, it is for example valid to place dieresis over the consonants π, τ and χ of the nasal

---

[3]personal communication

[4]Μανόλης Τριανταφυλλίδης, *Νεοελληνική Γραμματική της δημοτικής*, Ανατύπωση της έκδοσης του ΟΕΣΒ 1941 με διορθώσεις, Θεσσαλονίκη 2002, σελ. 66, ενότητα 133.

complexes μπ, ντ, γκ when it is necessary to show that these are pronounced, as written, voiceless, and not voiced. For example,

$$\text{κομπ̈ρέσα,} \quad \text{αντιαν̈ταν̈τικός,} \quad \text{ελεγ̈κτής}$$

(see previous footnote). The fonts support this writing if a combining dieresis is placed after the letter to receive it. The combining dieresis is the character U0308. On Linux desktops this is easily entered pressing Alt+Control+u, release them, and type the sequence 0308 and space.

More than that, in linguistics, they need to combine several accents above Greek letters. All this stacking of accents is supported by the fonts. For example, one can write

$$\tilde{\bar{α}} \quad \breve{\bar{ε}} \quad \breve{Ᾱ} \quad \bar{\tilde{Α}} \quad \tilde{Ω̣} \quad \acute{\bar{α}} \quad \grave{\breve{Χ}} \quad \acute{\ddot{\bar{ι}}}$$

by placing the combining accents from the Unicode block U0300–U0362 plus the usual Greek accents *after* the letter. So the above was typed as

$$\text{α}^{-\,\sim} \quad \text{ε}^{'\,\sim} \quad \text{Α}^{\sim} \quad \text{Α}^{\sim\,-} \quad \text{Ω}^{'\,\sim} \quad \text{α}^{-\,'} \quad \text{Χ}^{\sim\,'} \quad \text{ι}^{-\,\ddot\,'}$$

4.3   Archaic Greek writing

The Sans Serif Regular font provides access to 6th century bce and 4th century bce Greek capitals in ss04 and ss03 lookups. The `fontsetup` package provides commands such as

`\textivbce{}, \ivbce, \textvibce{}` and `\vibce`

6th century bce:
ⲘΘΔΕΙΣ ΑΓΕΩⲘΕΤΡΒΤΟΣ ΕΙΣΙΤΩ

4th century bce:
ΜΗΔΕΙϚ ΑΓΕΩΜΕΤΡΗΤΟϚ ΕΙϚΙΤΩ

Moreover, all fonts (except Mono & Math) support Ancient Greek Numerals (the full Unicode block of Greek digits U10140–U1018E is supported), with most symbols designed from scratch (and did not exist in C. Beccari's original fonts). A few of the new symbols:

$$Ⱶ 𐅆 Ⲭ 𐅇 Ⱶ 𐅈 ϙ𐅵$$

The four numerals that already existed in this range (that is U10144–U10147) in Beccari's fonts have been altered to a new design matching the style of cm but also provide some Ancient Greek flair. The new designs in Serifed and SansSerifed are:

$$𐅄 𐅅 𐅆 𐅇 \qquad 𐅄 𐅅 𐅆 𐅇$$

The `fontsetup` package provides commands for all of the above symbols. The commands follow the Unicode name of each slot (minus the "Greek Acrophonic"). So the Unicode slot U1014F named "Greek Acrophonic Attic Five Staters" can be accessed with the command `\atticfivestaters` and it gives 𐅏; and the slot u10182 named "Greek Kyathos Base Sign" can be accessed with the command `\greekkyathosbasesign` and it gives 𐆂.

4.4 Support for Papyrology

Papyrology needs to declare that a glyph is missing from the papyrus or the papyrus is worn at this point and the papyrologist adds the missing glyph but it is not clear from the papyrus. This is done by adding a dot below the glyph and it is supported for all Greek glyphs in the upright fonts monotonic or polytonic:

$$Ạ \quad Ἀ̣ \quad Ἤ̣ \quad ἄ̣ \quad ῷ̣ \quad ῥ̣$$

where in the source we just typed the dot below (char U0323) after the glyph. This feature is supported for the 4th bce and 6th bce Greek in Sans:

$$ΓΕΩΜΕΤΡΙΑ \quad ΓΕΩΜΕΤΡΙΑ$$

4.5 Support for Chemistry

It happens often that Greek upright characters are needed in Chemistry. People often have trouble with this (and this is why packages such as `chemgreek` exist). If Greek keyboard is available then it is easy; you just type in Greek, say `β-glucan` to get "β-glucan". But many writers do not have the Greek keyboard enabled, and they do not need to. Usually they type `$\beta$-glucan` but the result "β-glucan" is not satisfying. One can use the "up" versions typing `$\upbeta$-glucan` but still the result "β-glucan" looks more Math than Chemistry. To help with this, the `fontsetup` package provides commands such as `\chemAlpha`, `\chemalpha`, `\chemBeta`, `\chembeta`, etc. So this information essentially would only belong to the `fontsetup` documentation if it was not for kappa and rho. If we type in Greek `κ-compound` we get "ϰ-compound" which is not satisfying, as kappa is too cursive for this use. So the NewCM family provides an alternative kappa for this reason and this is how `\chemkappa` is defined in `fontsetup`:

    \newcommand{\chemkappa}{\textrm{\char"03F0}}:

  We write `\chemkappa-compound` and now get "κ-compound".

(The `\textrm` command in the above definition is there to make the command work in math mode too.) Similar is the situation for `\chemrho` (ρ) and `\chemrhoalt` (ϱ).

5 Russian

Russian is supported using the glyphs from the `cmu` package but it has considerable improvements (for example, the quality of the bold sans (see below)).

> Я помню чудное мгновенье:
> Передо мной явилась ты,
> Как мимолетное виденье,
> Как гений чистой красоты.
>   (Пушкинъ)

Again, as in Greek there is a different kind of guillemots for Russian which are available in CharacterVariant 3 (cv03). Compare:

Defaults guillemots: «»   Russian guillemots: «»   Greek guillemots: «»

Same is the situation with Russian emdash which is shorter than the default:

Default emdash: —

Russian emdash: —

6 Hebrew

The Hebrew blocks U0590–U05FF and Hebrew Presentation forms UFB1D–UFB4F are fully covered. and A few letters from Hebrew:

אבגדהושׁשׂלצ

7 Coptic and Epact Numbers

The Coptic language is fully supported. This covers the Coptic blocks in the Greek and Coptic Unicode block (U03E2–U03EF), the full Coptic Unicode block (U2C80–U2CFF) and the Coptic Epact Numbers (U102E0–U102FF). A few letters from Coptic and Epact numbers follow:

ⲗⲟⲅⲟⲥ ⲛ̀ⲁⲓⲅⲩⲡⲧⲓⲟⲥ   𐋂 𐋃 𐋄 𐋉 𐋊

8 Cherokee

Both Unicode blocks U13A0–13FF and UAB70–UABBF for Cherokee are supported. A few letters are:

ᎦᎣᏤᎢᏁᏇ ᎦᏫᎧᎦᏣ

9 Medieval Latin and Uncial Greek

The family includes new shapes for medieval and uncial Greek. The `fontsetup` package provides `\textuncial{<text>}` and `{\uncial text}` to use this shape. Let us write a sentence in this shape:

> Τhe Ρψτhαγορεαν Τheoρεμ is oνe οϝ τhe μοστ ιμπορταντ τheoρεμs iν Ϻατhεματιcs.
>
> Τo Πψθαγορειo Θεωρημα ειναι απο τα chμαντικοτερα θεωρηματα των Ϻαθηματικων.

10 Braille

Braille, both 6dot (uni2801–uni283F) as well as 8dit (uni2840–uni28FF) patterns are included in two versions. The Regular font provides the characters for sighted persons (such as teachers) so they can easily see which dots are on and which off. The Sans font contains the true Braille characters. I decided to have the sighted version in the Regular font since a blind person does not need the real Braille pattern, as those are produced by embossers. The Braille patterns here are meant as fonts to typeset text mainly for sighted persons.

	6dot	8dot
Regular version	⠃⠑⠊⠕⠥	⠃⠑⠊⠕⠥
Sans version	⠃⠑⠊⠕⠥	⠃⠑⠊⠕⠥

11 Ipa symbols

IPA symbols are included and following a suggestion of Huanyu Liu the kerning found in `tipa` package has been added here and further improved. Moreover the letters eth, eng, beta, theta and chi exists in IPA-style in the fonts and are accessible in the ss05 lookup table since they are in a different design from the Latin and Greek letters. You can access this lookup table using the `\textipa` command of the fontsetup package.

	Non-IPA	IPA
Regular	ð ŋ β ϑ χ	ð ŋ β θ χ
Sans	ð ŋ β θ χ	ð ŋ β θ χ

The joining of two characters such as t͜s, d͜z, k͡p, t͡ʃ etc is also supported. The low tie is the character U035C and the upper tie is U0361. The t͜s is produced by typing the sequence t then the low tie and then s. Similarly k͡p is produced by typing the sequence k then the upper tie and then p. For t͡ʃ and d͡ʒ the fonts have a contextual chaining substitution table that uses a tie which prints lower so it does not touch ʃ and ʒ. In cases that one wants to show the tie as x͜y͡z then

one needs to enable the cv5 character variant since the tie characters (U035C and U0361) are marks and not base glyphs. One can do that with a command such as

`\newfontfamily{\showtie}[CharacterVariant=5]{NewCM10-Book.otf}`

There is also an older practice that such sequences are joint into a ligature. This is not the modern way of writing but it seems than many people in the linguistics still prefer it. The fonts support this, if one enables the `lipa` table (local ipa) of the fonts. This can be done by adding the `RawFeature=+lipa` to the font specs when loading it and it is done automatically with the commands `\textoldipa{arg}` and `{\oldipatext arg}` of the `fontsetup` package.

For example, `\textoldipa{ts, tʃ}` produces ts, ʧ.

All other symbols of the `tipa` package are supported. Some examples are:

$$p͡, ', ɬ, ḷ, ɖ, ɟ, ṣ, ẹ, ḳ, ṃ, ṱ, ṭ, \text{etc.}$$

It is worth noting that all of the above is also available in the Mono family:

$$|t͡s|t͡ʃ|ʦ|p͡|'|ɬ|ḷ|ɖ|ɟ|ṣ|ẹ̣|ḳ|ṃ|ṱ|ṭ|$$

Next we give a real example of using IPA symbols. The quote below is from *A course in phonetics*, Ladefoged, Peter and Johnson, Keith, (2014), Cengage learning, pp. 285–286, and the `StylisticSet=05` has been enabled:

> Other examples of this phenomenon have also been mentioned. We saw that before [ŋ], English does not distinguish between tense and lax vowels. Consequently, the vowel that occurs in, for example, sing has a quality between that of [i] and [ɪ]. Similarly, there is no distinction between tense and lax vowels before [ɹ]. The vowel in here in most forms of American English is also intermediate between [i] and [ɪ]. The principle of maximum perceptual separation also accounts for some of the differences between languages. French has two high rounded vowels, [u] as in *tout* [t̪u] "all," and [y] as in *tu* [t̪y] "you."

I want to thank निरंजन (Niranjan) for suggesting and testing these improvements.

12 Bold Sans

`lm` fonts and `cmu` fonts do not contain a properly made BoldSans. Their Bold-Sans is a stroke-extension of the Sans with rounded corners. NewCM fixes that and provides a true BoldSans:

XΞ|XΞЯДЛ

13 Unicode Math coverage and options

NewCM provides full Unicode math support, that is all Mathematics Unicode Slots presented in `http://www.unicode.org/charts/` in both Math weights, Regular and Book. These blocks are:

Mathematical Symbols
> Arrows (uni2190–uni21FF)
> Supplemental Arrows-A (uni27F0–uni27FF)
> Supplemental Arrows-B (uni2900–uni297F)
> Supplemental Arrows-C (u1F800–u1F8FF)
> Additional Arrows (uni2B00–uni2BFF)
> Miscellaneous Symbols and Arrows (uni2B00–uni2BFF)

Mathematical Alphanumeric Symbols
> (u1D400–u1D7FF)
> Arabic Mathematical Alphabetic Symbols
> (u1EE00–u1EEFF)
> Letterlike Symbols (uni2100–uni214F)

Mathematical Operators
> (uni2200–uni22FF)
> Basic operators: Plus, Factorial
> (uni0000–uni007F)
> Division, Multiplication
> (uni0080–uni00FF)
> Supplemental Mathematical Operators
> (uni2A00–uni2AFF)
> Miscellaneous Mathematical Symbols-A
> (uni27C0–uni27EF)
> Miscellaneous Mathematical Symbols-B
> (uni2980–uni29FF)
> Floors and Ceilings (uni2308–uni230B)
> Invisible Operators (uni2061–uni2064)

Geometric Shapes (uni25A0–25FF)
> Additional Shapes (uni2B00–uni2BFF)
> Box Drawing (uni2500–uni257F)

Block Elements (uni2580–uni259F)
Geometric Shapes Extended (u1F780–u1F7FF)

Unfortunately, the `unicode-math` package does not provide commands currently for the hundrends of extra glyphs that have been added in order to fully cover the above Unicode ranges. The user can consult the Unicode charts at the above link and access the required glyph with `\char"#` where # is the Unicode number of the slot the glyph belongs to.

For example, `\char"2BDA` will give the Hygeia symbol (uni2BDA) the Rod of Asclepius as shown above (grayed and scaled ×8). The glyph that appeared in TUGboat (see [AT]), being more realistic will be moved to a new font in the future with ornaments.

13.1   Optical sizes for more glyphs

So far the fonts provided optical sizes for 1st and 2nd order exponents for letters. This was not true though for binary operators and some symbols commonly used in mathematics. For example, the `\perp` symbol ($\perp$) often appears in 2nd order exponents and then it appeared very thin. Some printers could even hardly print its thin lines. Now, such glyphs plus several binary operators, such as $+$, $-$, $\pm$, $\dotplus$ etc are now provided in optical sizes, so that expressions such as

$$+^{+^{+}} \quad \perp^{\perp^{\perp}} \quad *^{*^{*}} \quad \frac{|P_{F_{\perp}}(K)|}{e^{(x+y)^{*}}}$$

appear on screen and print properly on printers. Zoom or print and compare with `latinmodern-math` font:

$$+^{+^{+}} \quad \perp^{\perp^{\perp}} \quad *^{*^{*}} \quad \frac{|P_{F_{\perp}}(K)|}{e^{(x+y)^{*}}}$$

13.2   Upright and extensible integrals

The Math fonts (both Regular and Book weights) include upright integrals in the ss02 StylisticSet. Use with

`\setmathfont[StylisticSet=2]{NewCMMath-Book.otf}`
or
`\setmathfont[StylisticSet=2]{NewCMMath-Regular.otf}`

or use the `upint` option of the `fontsetup` package with

`\usepackage[upint,default]{fontsetup}`

for the Book weight, or

`\usepackage[upint,olddefault]{fontsetup}`

14

for the regular weight.

Moreover, extensible integrals are supported by the fonts but *NOT* by the Unicode TeX engines. The following code is a trick so that extensible integrals can be constructed using LuaLaTeX. The result is shown at the end of the article. What the code below does, is that it defines the slot uni222B (integral) as a delimiter. And then this is extended as a delimiter with the mechanism that the font provides.

```
\documentclass{article}
\usepackage[default]{fontsetup}
\begin{document}
$
\Uleft\Udelimiter 0 0 "222B
\begin{pmatrix}
  1\\2\\3\\4\\5\\6\\7\\8\\9
\end{pmatrix}
\Uright.
$
\end{document}
```

$$\int \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{pmatrix}$$

13.3 Additional and alternative characters in Math

The Math fonts provide the character `\varnothing` ($\varnothing$), as an alternative to `\emptyset` (a slashed zero), through Character Variant `cv01`. The `fontsetup` package provides the option '`varnothing`' to easily switch to the alternative character.

It also provides two more symbols that correspond to the commands

`\nrightrightarrows` ($⇉̸$)

and

`\nleftleftarrows` ($⇇̸$)

and supported by the `default` and `olddefault` options of the `fontsetup` package. These symbols are not in the Unicode Standard and so they are added in the Private Area of the fonts.

14 The Medieval Latin and Uncial Greek glyph complement

Table 1: NewCMUncial10-Book.otf

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Table 1: NewCMUncial10-Book.otf *cont.*

Basic Latin

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+0020 – 002F		!	"	#	$	%	Œ	'	(	)	*	+	,	-	.	-
U+0030 – 003F	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	-
U+0040 – 004F	@	A	B	C	D	Є	F	G	Ħ	I	J	K	L	M	N	O
U+0050 – 005F	P	Q	R	S	T	U	V	W	X	Y	Z	-	-	-	-	-
U+0060 – 006F	-	a	b	c	d	e	f	g	h	i	j	k	L	m	n	o
U+0070 – 007F	p	q	r	s	t	u	υ	w	x	y	z	-	-	-	-	-

Latin-1 Supplement

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+00A0 – 00AF	-	-	-	-	-	-	-	-	¨	-	-	-	-	-	-	-
U+00B0 – 00BF	-	-	-	-	-	-	-	·	-	-	-	-	-	-	-	-

Greek and Coptic

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+0370 – 037F	-	-	-	-	ʹ	͵	-	-	-	-	-	-	-	-	;	-
U+0380 – 038F	-	-	-	-	´	῁	Λ	·	Є	Η	Ι	-	Ο	-	Υ	ω
U+0390 – 039F	ϊ	Λ	Β	Γ	Δ	Є	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ʒ	Ο
U+03A0 – 03AF	Π	Ρ	-	C	Τ	Υ	Φ	Χ	Ψ	ω	Ϊ	Ϋ	λ	є	н	ι
U+03B0 – 03BF	ϋ̈	λ	в	г	λ	є	ʒ	н	θ	ι	κ	λ	μ	N	ʒ	o
U+03C0 – 03CF	π	ρ	c	c	τ	γ	Φ	χ	Ψ	ω	ϊ	ϋ̈	o	γ	ω	-
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Total number of glyphs in NewCMUncial10-Book.otf: 160

References

[AT] Antonis Tsolomitis, *The NewComputerModern font family*, TUGboat Vol. 42, No. 1, 2021.