

Chapter 8

Fonts and Typography

8.1 Fonts

A type “font” is a collection of letters, numbers and punctuation marks which are designed to blend with each other so as to produce text that is easy to read and aesthetically pleasing. A huge number of fonts are available on modern computers and typesetting equipment.

Book designers and graphic artists lie awake at night, thinking about the proper choice of font for their next project. Why do engineers and scientists have to worry about this stuff?

The reason is that every graph or animation is a blend of text and graphical objects which must blend together to enhance, rather than confuse, the meaning. In short, every scientific plot is a graphical design.

This in turn reflects a practical challenge: every program for making a graph offers choices of font type, font size and font style. It is helpful to understand what all these choices *mean*.

8.2 Type Dimensions

8.2.1 Monospaced and Proportional Type

Typewriters, because of the limitations of their carriage mechanisms, were forced to shift the same amount of space to the right for every letter or space. Typewriters therefore use fonts in which each letter has the same width.¹ (The two standard widths were “pica” or “ten-pitch”, which is ten characters per inch, or “elite” or “twelve-pitch”, which is twelve characters per inch.) Such fonts are called “monospaced” fonts. Similarly, the first generation of personal computers allocated the same number of pixels to each letter, and used monospaced fonts, too.

Printers learned rather early that monospaced fonts look ugly. Why use the same width for “m”, which has three vertical lines, as for “i”, which has just one? Since typesetters composed type by dropping individual letters into a frame, different widths for different letters caused no problems. Such variable-width collections of letters are called “proportional” fonts because the width of each letter on the page is the width each letter needs.

¹So-called “executive” typewriters could generate proportionally-spaced text, but these were extremely pricey and rare.

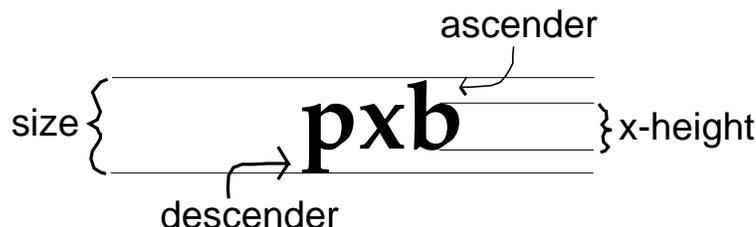


Figure 8.1: The size of a font in points is the vertical height, measured in units of $1/72$ of an inch, of the bottom of a “descender” to the top of the “ascender”

On late twentieth century workstations, almost all fonts are proportional. The only two monospace fonts that are still in wide use are “Courier” and “Monaco”. (The former was the usual font on business typewriters.)

Monospace fonts have survived, barely, for several reasons:

1. Improves accuracy of OCR (Optical Character Recognition); good for paper archives.
2. Simplifies vertical alignment of numerical tables (obsolete).
3. Convenient for distinguishing computer code listings from the main text.

Since the earliest computer fonts were monospaced, there is a sort of historical tradition that computer listings are typeset in Courier or Monaco also. There is a practical reason that this tradition continues: horizontal indentation is a common trick for improving the readability of a code; all the statements of an inner loop are indented the same number of spaces from the left margin, for example. Again, monospaced fonts have the advantages of (i) vertical alignment and (ii) different font from the text.

The march of technology — better table-making facilities, simpler commands for the vertically-aligning the first letters of a line by tabs, etc — have caused monospaced fonts to continue to lose ground. Nevertheless, Courier and Monaco survive. In Latex, such fonts are called “typewriter” style, and Latex documents always have a default monospaced font available. It is accessed by typing

```
{\tt This is a monospaced font sample. 0 1 2 3 4 5 6 7 8 9}
```

```
This is a monospaced font sample. 0 1 2 3 4 5 6 7 8 9
```

8.2.2 Font Sizes

Type sizes are usually specified in “points” where (i) a point is $1/72$ of an inch and (ii) the size of a font is the *height* between the bottom of a descender and the height of an ascender as illustrated in Fig. 8.1. The reason that this ascender-to-descender height is used to measure type sizes is that the vertical lengths of the ascenders and descenders are the same for all letters in a given font. The other important and universal characteristic of a font is the “x-height”, which is defined to be the height of the letter “x”, and is the height of that part of a letter that excludes the ascender and descender.

For proportionally-spaced fonts, the *width* is *different* for each letter in the font — only the *heights* are the *same*. This is why font sizes are always quoted in terms of the height of the font. Different 12-point fonts, such as 12-point Times and 12-point New York, may have quite different widths.

The New York font is the default on Macintosh computers because it is highly readable on the screen. However, proposal writers who are trying to fit a wordy proposal

into the fifteen page limit of the National Science Foundation have learned that the best way to squeeze text is to use the Times font: among the standard Macintosh fonts, it is the narrowest. (Historically, Times, which is also known as “Times Roman”, was designed in the early twentieth century for the *London Times*, which wanted a narrow, dense font that would still be readable at small sizes.)

Different fonts of a given size may also have different “x-heights”. Many modern fonts have fairly large x-heights, which makes the letters appear bigger on the page. The different choices of width and x-height imply that a given block of text may occupy quite different amounts of space, depending on whether it is typeset in Times or New York or Palatino or whatever.

The text in books and journals is usually set in either 12-point, 11-point or 10-point text. In LaTeX, for example, the standard style files allow only these three sizes as the default size for an article.

Scientists and engineers often use larger font sizes for two reasons. First, graphs are often reduce by 50% is lettered in a 12-point and then reduced by a factor of two in each dimension, the published graph will be labelled in 6-point type. Only the most eagle-eyed reader will be able to read the letters. When a graph is to be reduced on publication, one should use 18-point or 24-point type to produce the graph. Often, it improves readability to use a larger-than-12-point size even if the graph is not going to be reduced.

Second, 12-point letters on a transparency are almost unreadable in a lecture hall unless the audience is equipped with binoculars! Not only graph-letters but *all* letters on a transparency should be printed in 24-point size, or perhaps even bigger!

Latex has a couple of ways to make large-sized text. First, there is a program called “Slitex” which is specifically designed for making slides. However, it is rather awkward to and seems to be sliding rapidly into obsolescence.

An easier strategy is to use the place the command

```
\begin{Huge}
```

at the beginning of the transparency text and

```
\end{Huge}
```

at the end of the block of text. The result is illustrated in the next paragraph:

This is a sample of Huge text; good for making transparencies

An alternative is the LARGE environment:

This is a sample of LARGE text; note that the name of the environment is all in capital letters.

Very small point sizes (9 points or smaller) are sometimes used to label graphs when (i) the graph will not be reduced by the publisher and (ii) there is no room for larger text. Numerical labels for contour lines, for example, often require small text if there are a lot of lines. However, this is a desperate remedy because small type may not be legible in published form.

8.3 Serifs

Most fonts used for the body of a text are so-called “serif” fonts. The “serifs” are the little points at the ends of the lines as shown in Fig. 8.2. Serifs were first observed on Roman stone inscriptions as wedge-shaped indentations at the edge of the strokes. Were they convenient for the stone-cutters? Or were they simply decorative flourishes?

No one knows. What is clear is that serifs improve legibility. And so serifs can be found in the semi-uncial letters used in the Latin church between the fourth and ninth century, in the Book of Kells (Ireland, around 800 A. D.) in the Caroline miniscule standardized by Charlemagne’s scribes throughout his empire, and finally in the scripts of late medieval scribes.

In 1465, two German printers named Conrad Sweynheim and Arnold Pannartz were invited by Cardinal Turrecremata to come work for him at the Benedictine monastery of Subiaco. Rediscovering Carolingian manuscripts in the monastery library and also reading manuscripts in the flowing, humanistic script favored by Italian scribes, Sweynheim and Pannartz rejected the “Gothic” or “black letter” fonts employed by Gutenberg and designed the first printed Roman alphabet. The black letters, they thought, were the degenerate product of the barbarians that destroyed Rome. So, they attempted to return to the purity of their models.

The only problem was that the Carolingian minuscules didn’t look much like the Roman stone inscriptions (except that both had serifs!) Sweynheim and Pannartz solved this problem by creating the first double alphabet: capital letters modelled on the Roman stone inscriptions, small letters based on Carolingian miniscule. One of their ways of reconciling these two different styles was to add additional serifs to the miniscule letters. Pannartz died in 1476 and Sweynheim only a year later. However, all modern text fonts in English and most European languages are the descendents of their types: uppercase and lowercase letters with serifs.

Although modern printers are no longer worried about restoring the lost glories of Rome, serifs and uppercase/lowercase have survived for a very pragmatic reason: they improve legibility. The reason they improve legibility is that these devices increase the contrast between different letters and thereby make it easier for the eye and mind to quickly identify the identities of the letters and concentrate on extracting the meaning. This must be a paramount goal of scientific visualization, too.

Fonts that lack serifs are called “sans serif”. Such fonts are rarely used for long blocks of text because the absence of serifs causes eye fatigue.

Sans serif fonts, however, are fine for headlines, section headers, and most important for scientific visualization, for graph labels and legends. In single words or short lines, sans serifs fonts are not tiring to the eye. The absence of serifs means that the basic strokes of the letter can be made thicker and stronger. This is good in two very



Figure 8.2: A comparison of a serif font with a sans serif font. The small circles surround serifs.

different situations.

First, at large size, sans serif fonts are good for headlines because their thick strokes and different style makes the headline stand out from the body of the text. At small size, the thick strokes and absence of small features, i. e., serifs, makes sans serif letters easier to read. Since graphs are drastically reduced for publication, and the letters and numbers must be fit in small spaces between curves or on axes, this legibility-at-small-size is a real plus for graph labels.

Fig. 8.3 compares three different fonts which show the differences between them due to the two major properties of fonts: serif/sans serif and proportionally-spaced/monospace.

Serif, Proportionally Spaced: Times Roman
Sans Serif, Proportionally Spaced: Helvetic
Serif, Monospaced: Courier

ghijklmnop	HIJ
ghijklmnop	HIJ
ghijklmnop	HIJ

Figure 8.3: A comparison of three different fonts, illustrating the two most important distinguishing properties: serifs (or the absence thereof) and spacing (proportional or monospaced).

8.4 Fonts and Graphs

Graphs in a book or journal are usually reduced in size so as to fill only a fraction of a page. In a Latex document, the reduction is usually accomplished under the author's control by specifying the width of the figure in a `beginfigure` specification. The reducibility test is: print the figure at the size required by the journal and inspect the letter for legibility.

If the figure is to be submitted the old-fashioned way, a drawing on paper which is then reproduced and simultaneously reduced by the publisher, then a good empirical procedure is Cleveland's Copy-of-a-Copy-Four-Ninths Rule. He suggests making a xerox of the figure at $(2/3)$ reduction, then making a similar copy of the xerox so that the final xerox is a copy-of-a-copy at $(4/9)$ the height and width of the original. If this twice-duplicated-and-reduced image is still legible, then your figure is probably proof against any reasonable reduction-and-reproduction procedures of the publisher.

Because original drawings produced in Matlab, Adobe Illustrator or the like as 15 x 15 cm images will be reduced to roughly half their original size and because 9 point type is the smallest that is easily read, it is best to use 18 point or 24 point fonts in a graph or drawing program wherever possible.

For normal reading, fonts with serifs are best. However, many drawing programs and Matlab typically employ sans serif fonts as the default. (Helvetica is the most common font). There are two reasons why sans serif is popular for graphical lettering. The first is that the absence of serifs is tiring when reading large blocks of text. However, graph labels are so short that fatigue is not a problem. The second reason is that sans serif fonts, because their letters are simpler than those of serif fonts, are easier to read at very small size or when the type has been smeared or damaged by inadequate reproduction. Sans serif fonts are good for graph lettering because reduction to a small size and poor reproduction are very common problems for published graphs.

However, the first rule is the most important: use a sufficiently large font size so that is still legible after reduction by the printer. If the letters and numerals are sufficiently large, one can use any font, with or without serifs, in a graph.

In summary:

1. **Use large font sizes (18-pt, 24-pt) for axis labels and other text in a graph.**
2. **Sans serif fonts like Helvetica are good in graphs**

8.5 Fonts in LaTeX

In \LaTeX , the document class command specifies three font *families* as follows:

1. document font family (almost always a Roman font)
2. sans serif font family
3. typewriter style font family

The default fonts in \LaTeX are the Computer Modern fonts developed by Donald Knuth, the designer and programmer of \TeX . He designed a Roman font family, a sans serif Computer Modern, and a typewriter style font family. He also created a special italicized math font. This is employed only in the “math environment”. Knuth designed these four font families so they would all go well together, visually.

Because \LaTeX and \TeX needs a lot of information about fonts — not merely the fonts themselves, as required by printer and screen display, but also extra files with the so-called “font metrics” are needed, too — there is rather limited freedom to change fonts in these programs. Non-Postscript fonts may display and print okay in other software; I have amused myself on my Macintosh by using a Celtic font, created by a hobbyist, downloaded from the Internet. The Celtic letters displayed and printed just fine in Microsoft Word, but could not be used in \LaTeX or \TeX .

For general purpose math fonts, there seem to be only three options:

1. Computer Modern
2. MathTime
3. Lucida Bright

The Computer Modern fonts are in the public domain and are *always* included with any and all distributions of \LaTeX . The MathTime and Lucida Bright fonts must be purchased from firms such as Y & Y, Inc., and Blue Sky, Inc. The MathTime font was developed by the mathematician Michael Spivak to better match the Times Roman font. The Lucida Bright math fonts are always sold and used as a set with Lucida Bright text fonts

Knuth is a great computer scientist but only fair as a font designer. The Computer Modern fonts unfortunately are composed of rather thin strokes which don’t reproduce too well on a xerox machine — always readable, but hard on the eyes. The MathTime and Lucida Bright fonts are thicker and easier to read than Knuth’s fonts.

There is a somewhat larger range of text fonts possible in \LaTeX . In particular, the LaTeX command

```
\renewcommand{\familydefault}{pp1}
```

, placed before `\begin{document}`, changes the text fonts to Palatino. It is also necessary to add “palatino” [to activate “palatino.sty”] to the arguments of the `\usepackage`

The palatino font is better aesthetically than Computer Modern.

Similarly,

```
\renewcommand{\familydefault}{ptm}
```

changes the text fonts to Times, even if the commercial MathTime font has not been purchased.. It is also necessary to add “time” [to activate “time.sty”] to the arguments of the `\usepackage`. As noted earlier, the Times font can be a godsend for compressing research proposals and papers so that they fit within page limits; Times is the narrowest of the widely-used fonts.

8.6 Symbol Overloading

In most fields of science and mathematics, there are conventions about symbols. The argument of a function is “ x ” when real-valued and “ z ” when complex-valued. The set of integers is denoted by \mathbb{Z} , and so on. (The symbol \mathbb{Z} , with its doubled left vertical stroke, is an example of the so-called “Blackboard Bold” symbols that are part of standard LaTeX, UPPERCASE ONLY, and accessed through `\mathbb{Z}`, etc.)

The “mnemonics” principle is that it is easier to remember the meaning of symbols if there is a logical connection between the symbol and what it symbolizes. Good examples are using “ g ” to denote the gravitational constant, “ m ” or “ M ” for mass, and so on.

When an engineer is introducing new symbols, it is highly desirable to follow these two principles:

1. Respect the naming conventions of your field and
2. Employ mnemonics

Unfortunately, these principles sometimes lead to difficulties because the mnemonics principle suggests that we use a letter which is already taken for another quantity.

One remedy is to use Greek letters: if one needs a symbol which is “ g ”, and “ g ” is already taken, then γ . Unfortunately, many Roman letters do not have Greek counterparts, and some are reserved for standard symbols. Σ is the summation symbol and Γ is the factorial function, for example.

Standards LaTeX also supports the Hebrew letter “aleph” via `\aleph : \alpha`.

The free **amssymb** package adds three more letters: `\daleth` \aleph , `\beth` \beth , `\gimel` \beth .

A second strategy is to use accents. The LaTeX math mode accents are listed in Table 8.1.

The package “amsmath” allows three-dot and four-dot accents; “amsxtra” allows some additional accents not tabulated above.

The wide symbols and overbrace are not “accents” in the sense that a typesetter would understand. However, accents and more elaborate accentuations share the same problem: LaTeX and other typesetting systems are usually more sophisticated than the text schemes of illustration and image-processing programs. An overbrace, for example, can be created in Adobe Illustrator. It is usually necessary, however, to draw a brace, rotate it through ninety degrees, change the scale, and then manually place this over the letters it should span. The elements can be grouped so that the brace-plus-letters can then be moved and resized as a group. But one should be careful in using complicated symbols, and remember that there may be difficulties in reproducing them in an illustration program.

A third option is to use variant font symbols. The “calligraphic” alphabet, displayed in the second line below, is UPPERCASE ONLY and is built-in to LaTeX. The “Fraktur” alphabet uses black letter; it requires the “amfonts” package [available free by download from the American Mathematical Society web site]; is invoked by

```
\mathfrak{a}
```

, and is available in both lowercase and uppercase. The “Euler script” requires

```
\usepackage[mathscr]{euca}
```

, is invoked by

`\mathscr{A}`

, and is UPPERCASE ONLY. Boldface comes in two species: the standard or “upright” form, which is

`\mathbf{a}`

, and the “italic” form, which is

`\boldsymbol{a}`

. All boldface symbols are available in both uppercase, lowercase and numerals, etc.; the italic-bold requires the (free) “amsfonts” package.

Roman: *ABCDEFGHIJKLMNOPQRSTUVWXYZ*
 calligraphic: *ABCDEFGHIJKLMNOPQRSTUVWXYZ*
 Fraktur: *ABCDEFGHIJKLMNOPQRSTUVWXYZa*
 Euler Script: *ABCDEFGHIJKLMNOPQRSTUVWXYZ*
 Bold-upright: **ABCDEFGHIJKLMNOPQRSTUVWXYZa**
 Bold-Italic: ***ABCDEFGHIJKLMNOPQRSTUVWXYZa*** (8.1)

Calligraphic or Gothic letters pose similar problems to complicated accents. If one’s workstation lacks a black letter font, for example, one can scan in a copy of the needed

Table 8.1: Math Accents in Latex

Example		
\hat{a}	<code>\hat{a}</code>	circumflex
\tilde{a}	<code>\tilde{a}</code>	tilde
\acute{a}	<code>\acute{a}</code>	acute
\bar{a}	<code>\bar{a}</code>	macron or overbar
\breve{a}	<code>\breve{a}</code>	breve
\check{a}	<code>\check{a}</code>	caret or hacek or clicka
\dot{a}	<code>\dot{a}</code>	overdot
\ddot{a}	<code>\ddot{a}</code>	umlaut or diaeresis
\grave{a}	<code>\grave{a}</code>	grave
\vec{a}	<code>\vec{a}</code>	vector
\widehat{abc}	<code>\widehat{abc}</code>	wide circumflex
\widetilde{abc}	<code>\widetilde{abc}</code>	wide tilde
\overline{abc}	<code>\overline{abc}</code>	wide overbar
\underline{abc}	<code>\underline{abc}</code>	underline
\overbrace{abc}	<code>\overbrace{abc}</code>	overbrace
\overbrace{abc}^n	<code>\overbrace{abc}^n</code>	overbrace-with-superscript

character from a typography sample book, and then resize and duplicate the scanned image as needed. This is obviously much more difficult than simply typing a letter as an axis letter or part of a title. One sometimes sees labels such as “Gothic-g” as a way of honest labeling when the necessary font is available in Latex, but not in Illustrator or Matlab.

8.7 Dingbats, Ornaments and Pi Fonts

Fonts of symbols rather than letters are called “pi fonts” or “dingbat fonts”. The individual symbols are called “pi characters”, “dingbats” or “ornaments”. The latter term is more general; “printer’s ornaments” includes symbols that are not incorporated into fonts, but were in separate little woodcuts or metal plates that were inserted as ornamental borders, printer’s marks and so on.

The figures illustrate typical dingbat fonts. Some of the symbols, such as arrows, numbers in reverse type, and so on are useful in scientific graphs. The easiest way to insert such symbols is to insert a text block with a single character from the appropriate dingbat font. The text block can then be moved, duplicated and resized as necessary

Zapf Dingbats

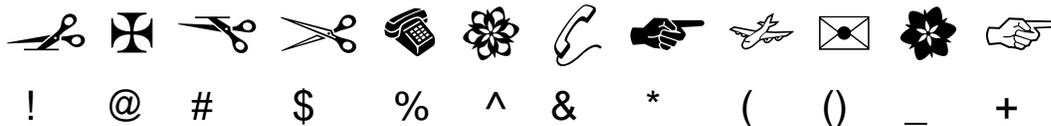
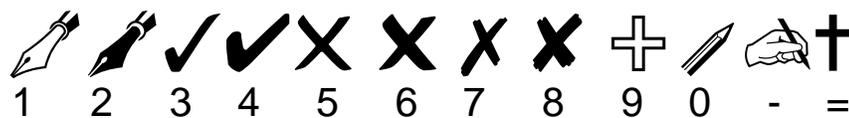
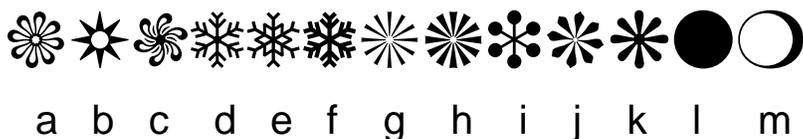


Figure 8.4: Symbols in the ITC Zapf Dingbat font, perhaps the most widely available “pi font”. The letters, numerals or punctuation marks that need to be typed to generate the symbol are shown immediately underneath it.

Webdings

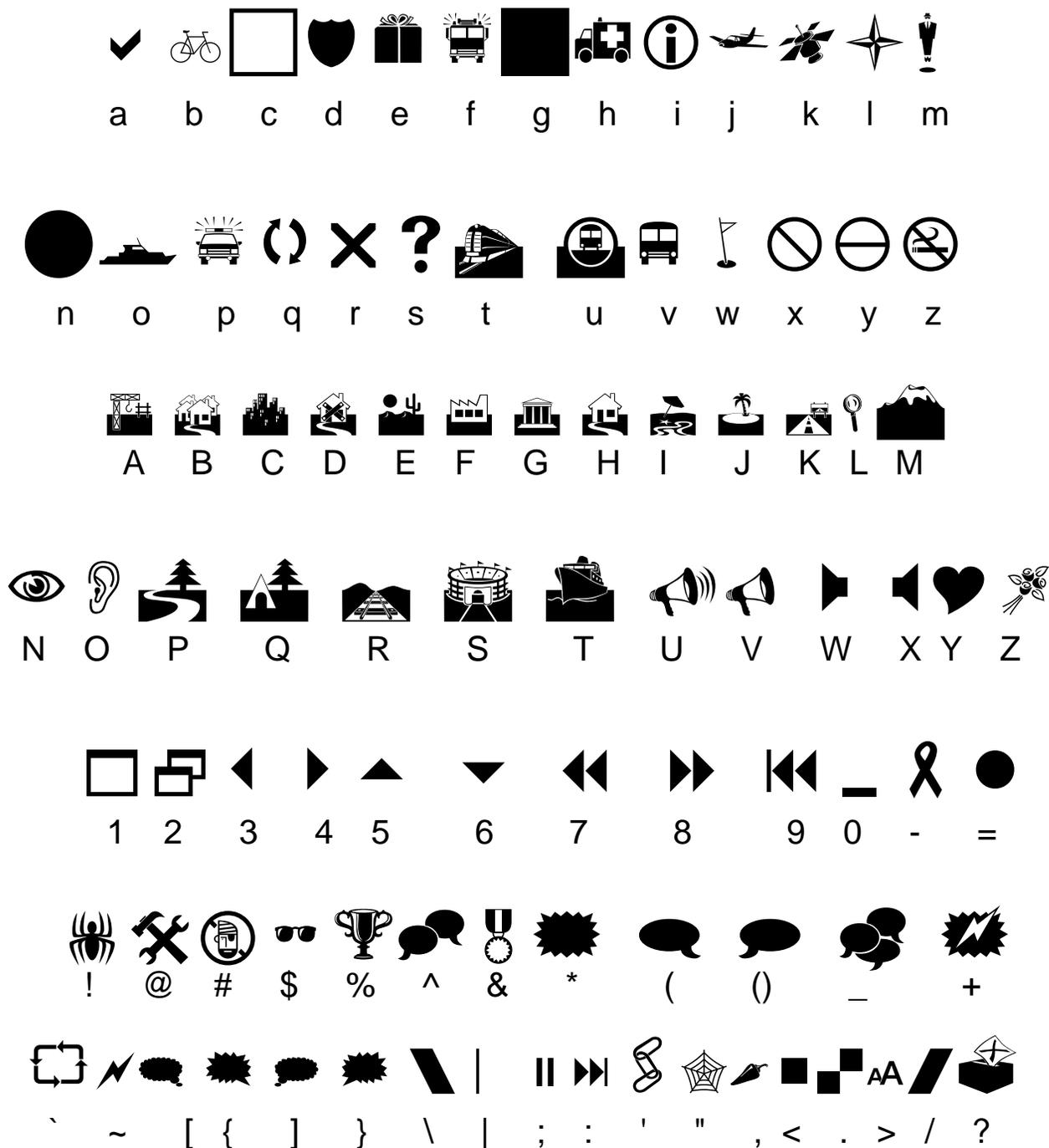


Figure 8.5: Symbols in the Webdings font. The letters, numerals or punctuation marks that need to be typed to generate the symbol are shown immediately underneath it.

Wingdings



a b c d e f g h i j k l m



n o p q r s t u v w x y z



A B C D E F G H I J K L M



N O P Q R S T U V W X Y Z



1 2 3 4 5 6 7 8 9 0 - =



! @ # \$ % ^ & * () - +



~ [{] } \ | ; : ' " , < . > / ?

Figure 8.6: Symbols in the Wingdings font. The letters, numerals or punctuation marks that need to be typed to generate the symbol are shown immediately underneath it.

Wingdings2

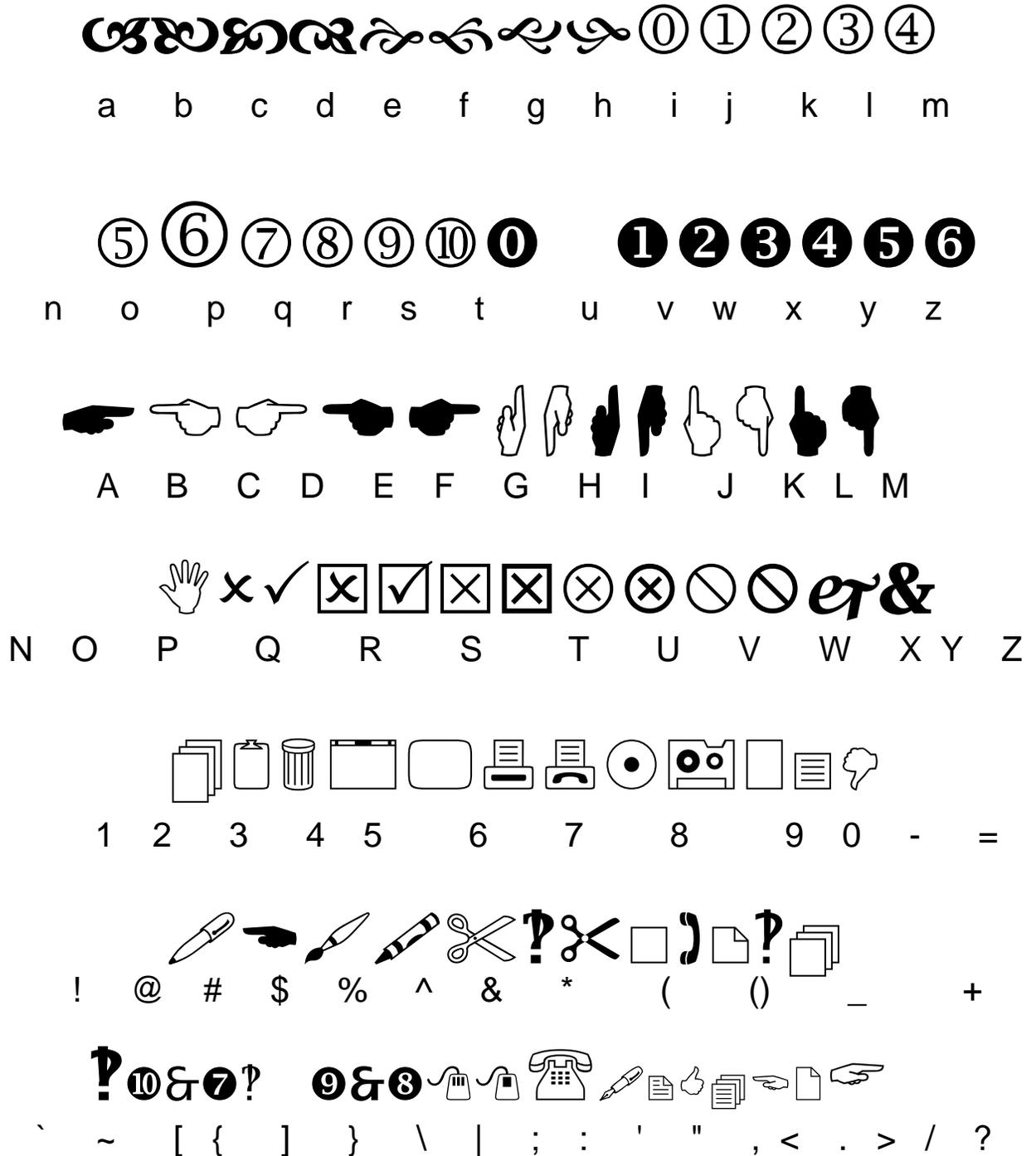


Figure 8.7: Symbols in the Wingdings2 font. The letters, numerals or punctuation marks that need to be typed to generate the symbol are shown immediately underneath it.

