

1

¹ Plain `TeX` defines the various `\big` commands by using `\left` and `\right` to provide a delimiter context. It sets the size by constructing an empty formula with the desired height.

The delimiters (except for ‘(’, ‘)’, and ‘/’) are among the symbols listed on pages ‘\lbrace’–‘\Uparrow’. They are listed in one place on page 146 of *The T_EXbook*.

A delimiter can belong to any class. For a delimiter that you enlarge with `\bigl`, `\bigr`, etc., the class is determined by the command: “opener” for l-commands, “closer” for r-commands, “relation” for m-commands, and “ordinary symbol” for g-commands, e.g., `\Big`.

You can obtain a delimiter in two different ways:

- 1) You can make a character be a delimiter by assigning it a non-negative delimiter code (see below) with the `\delcode` command (p. ‘\delcode’). Thereafter the character acts as a delimiter whenever you use it in a delimiter context.²
- 2) You can produce a delimiter explicitly with the `\delimiter` command (p. ‘\delimiter’), in analogy to the way that you can produce an ordinary character with the `\char` command or a math character with the `\mathchar` command. The `\delimiter` command uses the same delimiter codes that are used in a `\delcode` table entry, but with an extra digit in front to indicate a class. It’s rare to use `\delimiter` outside of a macro definition.

A delimiter code tells T_EX how to search for an appropriate output character to represent a delimiter. The rules for this search are rather complicated (see pages 156 and 442 of *The T_EXbook*). A complete understanding of these rules requires knowing about the organization of font metrics files, a topic that is not just beyond the scope of this book but beyond the scope of *The T_EXbook* as well.

In essence the search works like this. The delimiter code specifies a “small” output character and a “large” output character by providing a font position and a font family for each (see p. ‘\delcode’). Using this information, T_EX can find (or construct) larger and larger versions of the delimiter. T_EX first tries different sizes (from small to large) of the “small” character in the “small” font and then different sizes (also from small to large) of the “large” character in the “large” font, seeking one whose height plus depth is sufficiently large. If none of the characters it finds are large enough, it uses the largest one that it finds. It’s possible that the small character, the large character, or both have been left unspecified (indicated by a zero in the appropriate part of the delimiter code). If only one character has been specified, T_EX uses that one. If neither has been specified, it replaces the delimiter by a space of width `\nulldelimiterspace`.

² It’s possible to use a character with a nonnegative delimiter code in a context where it isn’t a delimiter. In this case T_EX doesn’t perform the search; instead it just uses the character in the ordinary way (see page 156 of *The T_EXbook*).