1

```
\left
\right
```

These commands must be used together in the pattern:

$\texttt{\textbackslash left}\ \langle delim_1 \rangle\ \langle subformula \rangle\ \texttt{\textbackslash right}\ \langle delim_2 \rangle$

This construct causes TEX to produce $\langle subformula \rangle$, enclosed in the delimiters $\langle delim_1 \rangle$ and $\langle delim_2 \rangle$. The vertical size of the delimiter is adjusted to fit the vertical size (height plus depth) of $\langle subformula \rangle$. $\langle delim_1 \rangle$ and $\langle delim_2 \rangle$ need not correspond. For instance, you could use ']' as a left delimiter and '(' as a right delimiter in a single use of `\left` and `\right`.

`\left` and `\right` have the important property that they define a group, i.e., they act like left and right braces. This grouping property is particularly useful when you put `\over` (p. '`\over`') or a related command between `\left` and `\right`, since you don't need to put braces around the fraction constructed by `\over`.

If you want a left delimiter but not a right delimiter, you can use '.' in place of the delimiter you don't want and it will turn into empty space (of width `\nulldelimiterspace`).

Example:
```
$$\left\Vert\matrix{a&b\cr c&d\cr}\right\Vert
   \qquad \left\uparrow q_1\atop q_2\right.$$
```
produces:

$$\left\Vert\matrix{a & b\cr c & d\cr}\right\Vert \qquad \left\uparrow q_1\atop q_2\right.$$