

```
\loop  $\alpha$   $\beta$  \repeat
\repeat
```

These commands provide a looping construct for T_EX. Here α and β are arbitrary sequences of commands and is any of the conditional tests described in “Conditional tests” (p. ‘conds’). The `\repeat` replaces the `\fi` corresponding to the test, so you must not write an explicit `\fi` to terminate the test. Nor, unfortunately, can you associate an `\else` with the test. If you want to use the test in the opposite sense, you need to rearrange the test or define an auxiliary test with `\newif` (see above) and use that test in the sense you want (see the second example below).

T_EX expands `\loop` as follows:

- 1) α is expanded.
- 2) is performed. If the result is false, the loop is terminated.
- 3) β is expanded.
- 4) The cycle is repeated.

Example:

```
\count255 = 6
\loop
  \number\count255\
  \ifnum\count255 > 0
    \advance\count255 by -1
  \repeat
```

produces:

```
6 5 4 3 2 1 0
```

Example:

```
\newif\ifnotdone % \newif uses \count255 in its definition
\count255=6
\loop
  \number\count255\
  \ifnum\count255 < 1 \notdonefalse\else\notdonetrue\fi
  \ifnotdone
    \advance\count255 by -1
  \repeat
```

produces:

```
6 5 4 3 2 1 0
```