1

```
\lastkern
\lastskip
\lastpenalty
\lastbox
```

These control sequences yield the value of the last item on the current list.
They aren't true commands because they can only appear as part of an
argument. If the last item on the list isn't of the indicated type, they yield
a zero value (or an empty box, in the case of `\lastbox`). For example, if
the last item on the current list is a kern, `\lastkern` yields the dimension
of that kern; if it isn't a kern, it yields a dimension of 0.

Using `\lastbox` has the additional effect of removing the last box from
the list. If you want the original `\lastbox` to remain on the list, you have
to add a copy of it to the list. `\lastbox` is not permitted in a math list
or in the main vertical list.

These control sequences are most useful after macro calls that might
have inserted entities of the indicated kinds.

Example:
```
\def\a{two\kern 15pt}
one \a\a\hskip 2\lastkern three\par
% Get three times as much space before 'three'.
\def\a{\hbox{two}}
one \a
\setbox0 = \lastbox % Removes 'two'.
three \box0.
```
produces:

one two   two   three
one three two.