ED – exercise 04 – 2024.02.27

Editor

Goal

Iteration through a list. Forward, backward, removal, insertion… the works!

Instructions

Log into http://ed.fdi.ucm.es/domjudge/team with your credentials. Make sure to **submit at least 1 answer before the period ends**, even if it does not work. Before leaving the classroom, even if your program seems to work, **go by the teacher's desk** to see if it can be improved.

Problem statement

Instead of using a tried and tested text editor, your boss wants you to write your own, from scratch, relying only on a list implementation.

Text editors have text (initially empty) and an insertion position (initially the start of an empty list), which is where the cursor is displayed, and where characters will be added (if you press a normal key) or removed (if you press backspace or delete). Cursors can move the insertion position around, and there are two modes: in insert mode (the default), new characters are added at the current position; while in overwrite mode, new characters replace old ones. Note that text editors **do not throw exceptions** for illegal operations – they simply ignore them.

Input

Input is a series of lines. Each line contains keystrokes that will be entered into your editor. There are only 7 special characters: all others should be inserted as-is:

- `>` and `<` move the current insertion position to the left and right, respectively
- `@` toggles between insertion-mode and overwrite-mode
- `{` and `}` jump to the start and end of the document, respectively
- `\` and `/` erase to the left (backspace) and right (delete), respectively

Output

Output a single line for each input line: the text that results after those keystrokes.

Example input & output

```
345>{12                                12345
I hate pointers{{>@>love               I love pointers
C++ is great!<\\\\\crazy complex/.     C++ is crazy complex
```

Additional guidance

You ***must*** use the provided template. Note that MList.h *has* bidirectional iterators & `rend()`.