

文本编辑器

题目描述

现在需要你实现一个带光标的文本编辑器，需要具有以下功能：

- 添加：在光标处添加文本
- 删除：删除光标左侧的文本
- 移动：将光标向左向右移动
- 复制：复制光标左侧的文本
- 粘贴：在光标处添加上一次复制的文本

请你实现TextEditor类，代码模板如下

TextEditor.h

```
#include <string>

class TextEditor {
public:
    TextEditor();

    void addText(const std::string& text);

    void deleteText(int length);

    void moveCursor(int steps);

    void copy(int length);

    void paste();

    void print();
};
```

TextEditor.cpp

```
#include <TextEditor.h>

using namespace std;

TextEditor::TextEditor() {
    // TODO: 用空文本初始化
};

void TextEditor::addText(const string& text) {
    // TODO: 将 text 添加到光标所在位置
}
```

```
void TextEditor::deleteText(int length) {
    // TODO: 删除光标左边 length 个字符
}

void TextEditor::moveCursor(int steps) {
    // TODO: 将光标向左向右移动
}

void TextEditor::copy(int length) {
    // TODO: 复制光标左侧 length 个字符
}

void TextEditor::paste() {
    // TODO: 在光标处添加上一次复制的文本
}

void TextEditor::print() {
    // TODO: 输出当前编辑器内容
}
```

- `TextEditor()`: 用空文本初始化编辑器
- `void addText(const std::string& text)`: 在光标右侧添加文本`text`, 并将光标移动至所添加文本的右侧
- `void deleteText(int length)`: 删除光标左侧`length`个字符, `length`为正整数, 注意光标在最左端的情况
- `void moveCursor(int steps)`: `steps`为正数表示向右移动, 负数则表示向左移动, 注意光标无法超过文本的两端
- `void copy(int length)`: 复制光标左侧`length`个字符, 同样注意光标在最左端的情况
- `void paste()`: 在光标右侧添加上一次复制的文本
- `void print()`: 输出当前文本至控制台, 不需要对光标进行表示

推荐使用链表或对顶栈进行实现, 数组大概率TLE

数据约定

$1 < \text{text.length}, \text{length} < 100 - 10^6 < \text{steps} < 10^6$ 总操作次数不超过 $2 * 10^5$

测试样例

部分测试样例如下

`Main.cpp`留空即可

```
void TEST_0() {
    TextEditor editor;
    editor.addText("word");
    editor.moveCursor(-1);
    editor.addText("l");
    editor.moveCursor(-4);
}
```

```

    editor.addText("hello ");
    editor.print();
    editor.moveCursor(5);
    editor.deleteText(5);
    editor.addText("cpp");
    editor.print();
}
// hello world
// hello cpp

void TEST_1() {
    TextEditor editor;
    editor.addText("njuverygood");
    editor.copy(8);
    for (int i = 0; i < 5; i++) {
        editor.paste();
    }
    editor.print();
}
// njuverygoodverygoodverygoodverygoodverygoodverygood

void TEST_2() {
    TextEditor editor;
    editor.addText("hello");
    editor.moveCursor(-100);
    editor.copy(100);
    editor.moveCursor(100);
    editor.deleteText(100);
    editor.paste();
    editor.print();
}
// 输出为空行

```