

龙虾抄写器

Johannes Gutenberg 是一个德国的铁匠，他发明了便携式印刷术。人们都说 Leonardo 是 Johannes 的狂热追随者，因为 Leonardo 为了纪念 Johannes 而设计出一部非常精巧的打字设备，名叫龙虾抄写器(il gambero scrivano)。它有点像一部功能简单的现代打字机，它只接受二种指令：一个指令是打出下一个字符，而另一个指令则是取消最近的指令（或返回到最近的指令之前的状态）。这部龙虾抄写器的显著特点就是具有这个功能強大的取消指令：因为一个取消的指令本身也是一个命令，因此它也可以被取消。

说明

你的任务是实现该龙虾抄写器的软件版本，它由一个空白文本开始，并接收使用者输入的指令序列，并可查询目前文本中的特定位置。详细说明如下：

- Init() — 在执行的一开始被调用一次，没有任何参数。它可以被用来初始化数据结构，而且也绝不会被取消。
- TypeLetter(L) — 在文本的最后附加一个小写字母 L，此 L 可以是 a, ..., z。
- UndoCommands(U) — 取消最后的 U 条指令，U 是一个正整数。
- GetLetter(P) — 返回在目前文本中位置为 P 的字母，P 是一个非负整数的序号。文本中第一个字母的序号为 0(这个查询并不是一个指令，因此会被取消命令所忽略。)

在调用 Init()之后，上面的其它函数可以按任何顺序被调用 0 次或多次。题目保证 U 不会超过前面已经接收的指令数目，而且 P 一定小于当前文本的长度(在当前文本中的字母个数)。

对于指令 UndoCommands(U)，它会按照原来执行相反的顺序取消前面 U 个指令: 如果被取消的指令是 TypeLetter(L)，就会从文本的最后移除 L 个字母。如果被取消的指令是 UndoCommands(X)，那么会重新执行被取消的 X 个指令，并且要依照原来执行的顺序执行。

样例

我们列出一个调用序列，以及每次调用后的文本（结果）。

Call	返回	目前文字
Init()		
TypeLetter(a)		a
TypeLetter(b)		ab
GetLetter(1)	b	ab
TypeLetter(d)		abd
UndoCommands(2)		a
UndoCommands(1)		abd
GetLetter(2)	d	abd

TypeLetter(e) abde

UndoCommands(abd

1)

UndoCommands(ab

5)

TypeLetter(c) abc

GetLetter(2) c abc

UndoCommands(abd

2)

GetLetter(2) d abd

子任务 1[5 分]

指令和查询的总数在 1 到 100 之间(包括 100)，而且没有 UndoCommands 指令的调用。

子任务 2[7 分]

指令和查询的总数在 1 到 100 之间(包括 100)，而且没有 UndoCommands 指令被取消。

子任务 3[22 分]

指令和查询的总数在 1 到 5000 之间(包括 5000)。

子任务 4[26 分]

指令和查询的总数在 1 到 1000 000 之间(包括 1000 000)。所有对 GetLetter 的调用都将出现在对 TypeLetter 和 UndoCommands 的所有调用之后。

子任务 5[40 分]

指令和查询的总数在 1 到 1000 000 之间(包括 1000 000)。

实现细节

你所提交的只是一个文件，名为 `scrivener.c`, `scrivener.cpp` 或 `scrivener.pas`. 这个文件必须实现上述描述的子程序并使用下列程序行。

C/C++ 程序

以下几行程序参见英文

Pascal 程序

以下几行程序参见英文

这些子程序必须如上所描述。当然你可以自由实现其它子程序作为它们的内部使用。你的提交不能以任何方式与标准输入/输出交互，也不能与任何其它文件进行交互，。

样例评分程序

样例评分程序按照下列格式读入输入：

- ▮ 第一行为输入中的指令和查询的总数；

- ▮ 每一个后继行的格式：

对于 TypeLetter 的指令格式为 T 后跟一个空格和一个小写字母；

对于 UndoCommands 的指令格式为 U 后跟一个空格和一个整数；

对于 GetLetter 的指令格式为 P 后跟一个空格和一个整数；

样例评分程序将打印由 GetLetter 返回的字符，每个字符占用一个独立的行。