

代码编辑器	题目描述：
某公司为了更高效的编写代码，邀请你开发一款代码编辑器程序。程序的输入为 已有的代码文本和指令序列，程序需输出编辑后的最终文本。指针初始位置位于文本的开头。	
支持的指令( $X$ 为大于等于 $0$ 的整数, $word$ 为无空格的字符串):	
<b>FORWARD</b>	$X$ 指针向前(右)移动 $X$ , 如果指针移动位置超过了文本末尾, 则将指针移动到文本末尾
<b>BACKWARD</b>	$X$ 指针向后(左)移动 $X$ , 如果指针移动位置超过了文本开头, 则将指针移动到文本开头
<b>SEARCH-FORWARD</b>	$word$ 从指针当前位置向前查找 $word$ 并将指针移动到 $word$ 的起始位置, 如果未找到则保持不变
<b>SEARCH-BACKWARD</b>	$word$ 在文本中向后查找 $word$ 并将指针移动到 $word$ 的起始位置, 如果未找到则保持不变
<b>INSERT</b>	$word$ 在指针当前位置前插入 $word$ , 并将指针移动到 $word$ 的结尾
<b>REPLACE</b>	$word$ 在指针当前位置替换并插入字符 (删除原有字符, 并增加新的字符)
<b>DELETE</b>	$X$ 在指针位置删除 $X$ 个字符
输入描述：	
输入的第一行为命令列表的长度 $K$	
输入的第二行为文件中的原始文本	
接下来的 $K$ 行, 每行为一个指令	

输出描述：

编辑后的最终结果

补充说明：

文本最长长度不超过 256K

示例 1

输入：

```
1
ello
INSERT h
```

输出：

```
hello
```

说明：

在文本开头插入

示例 2

输入：

```
2
hllo
FORWARD 1
INSERT e
```

输出：

```
hello
```

说明：

在文本的第一个位置插入

示例 3

输入：

```
2
```

```
hell
```

```
FORWARD 1000
```

```
INSERT o
```

输出：

```
hello
```

说明：

在文本的结尾插入

示例 4

输入：

```
1
```

```
hello
```

```
REPLACE HELLO
```

输出：

```
HELLO
```

说明：

替换

示例 5

输入:

```
1
hello
REPLACE HELLO_WORLD
```

输出:

```
HELLO_WORLD
```

说明:

超过文本长度替换

示例 6

输入:

```
2
hell
FORWARD 10000
REPLACE O
```

输出:

```
hellO
```

说明:

超出文本长度替换

# 输入

```
K = int(input())
```

```
src = list(input())
```

```
item = []
```

```
for i in range(K):
```

```
    item.append(input().split(' '))
```

# 指针

point = 0

# 执行

for i in range(K):

if item[i][0]=="FORWARD":

move = int(item[i][1])

point = min(point+move,len(src))

elif item[i][0]=="BACKWARD":

move = int(item[i][1])

point=max(point-move,0)

elif item[i][0] == "SEARCH-FORWARD":

word = item[i][1]

index = ".join(src[point:]).find(word)

if index != -1:

point+=index

elif item[i][0] == "SEARCH-BACKWARD":

word = item[i][1]

index = ".join(src[:point]).find(word)

if index != -1:

point = index

elif item[i][0] == "INSERT":

```
word = item[i][1]

src[point:point] = list(word)

point = point+len(word)

elif item[i][0] == "REPLACE":

    word = item[i][1]

    src[point:point+len(word)]=list(word)

    point = point+len(word)

elif item[i][0] == "DELETE":

    x = int(item[i][1])

    src[point:x+point] = []

else:

    print(f'执行失败: {item[i][0]}')


print("".join(src))
```