

Máy soạn thảo

Có một số người nói rằng Leonardo thần tượng Johannes Gutenberg, là một thợ rèn người Đức, người đã phát minh ra máy đánh chữ. Leonardo đã bày tỏ lòng kính trọng bằng việc thiết kế ra cỗ máy được gọi là máy soạn thảo tôm đồng — *il gambero scrivano* — một thiết bị đánh máy đơn giản. Nó hoạt động gần giống các máy đánh chữ hiện đại và chỉ nhận 2 loại lệnh: lệnh để thực hiện gõ ký tự tiếp theo và lệnh để khôi phục các lệnh thực hiện ngay trước đó. Khả năng đáng lưu ý của máy soạn thảo này là câu lệnh khôi phục rất mạnh: lệnh khôi phục cũng được coi là một lệnh và nó có thể được khôi phục.

Phát biểu bài toán

Nhiệm vụ của bạn là cài đặt phiên bản phần mềm của máy đánh chữ tôm đồng: Nó sẽ bắt đầu từ văn bản rỗng và nhận một dãy lệnh của người sử dụng và các truy vấn đối với vị trí cụ thể của phiên bản hiện tại của văn bản như sau.

- `Init()` — được gọi một lần ở lúc bắt đầu thực hiện, không có tham số. Nó được sử dụng để khởi tạo các cấu trúc dữ liệu. Nó sẽ không khi nào bị khôi phục.
- `TypeLetter(L)` — thêm vào cuối văn bản một chữ cái in thường `L` được chọn từ `a, ..., z`.
- `UndoCommands(U)` — khôi phục lại `U` lệnh cuối cùng, trong đó `U` là một số nguyên dương.
- `GetLetter(P)` — trả lại ký tự ở vị trí `P` trong văn bản hiện tại, trong đó `P` là chỉ số nguyên không âm. Ký tự đầu tiên trong văn bản có chỉ số 0. (Truy vấn này không là câu lệnh và vì thế nó sẽ bị bỏ qua khi thực hiện lệnh khôi phục.)

Sau lệnh đầu tiên gọi `Init()`, các chương trình con khác có thể được gọi 0 hoặc nhiều lần theo thứ tự tùy ý. Bạn được đảm bảo là `U` sẽ không vượt quá số câu lệnh đã được thực hiện trước đó, và `P` sẽ nhỏ hơn độ dài của văn bản (số ký tự trong văn bản hiện tại).

Đối với lệnh khôi phục `UndoCommands(U)`, lệnh này khôi phục `U` câu lệnh trước đó theo thứ tự *ngược lại*: nếu câu lệnh cần khôi phục là `TypeLetter(L)`, thì nó sẽ loại bỏ `L` từ cuối văn bản hiện tại; nếu câu lệnh cần khôi phục là `UndoCommands(X)` với một giá trị `X` nào đó, nó sẽ thực hiện lại dãy `X` câu lệnh trước đó theo đúng thứ tự *ban đầu* của chúng.

Ví dụ

Chúng ta trình bày một dãy các lệnh cùng với trạng thái của văn bản sau mỗi lệnh.

Lệnh gọi	Trả lại	Văn bản hiện thời
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(1)		abd
UndoCommands(5)		ab
TypeLetter(c)		abc
GetLetter(2)	c	abc
UndoCommands(2)		abd
GetLetter(2)	d	abd

Subtask 1 [5 điểm]

- Tổng số câu lệnh và truy vấn nằm giữa 1 và 100 (kể cả hai đầu mút) và không có lệnh gọi `UndoCommands`.

Subtask 2 [7 điểm]

- Tổng số câu lệnh và truy vấn nằm giữa 1 và 100 (kể cả hai đầu mút) và không có lệnh `UndoCommands` nào được khôi phục.

Subtask 3 [22 điểm]

- Tổng số câu lệnh và truy vấn nằm giữa 1 và 5 000 (kể cả hai đầu mút).

Subtask 4 [26 điểm]

- Tổng số câu lệnh và truy vấn nằm giữa 1 và 1 000 000 (kể cả hai đầu mút). Tất cả các lệnh gọi `GetLetter` đều xuất hiện sau tất cả các lệnh gọi `TypeLetter` và `UndoCommands`.

Subtask 5 [40 điểm]

- Tổng số câu lệnh và truy vấn nằm giữa 1 và 1 000 000 (kể cả hai đầu mút).

Chi tiết cài đặt

Bạn phải giao nộp một file có tên là `scrivener.c`, `scrivener.cpp` hoặc `scrivener.pas`. File này phải cài đặt các chương trình con mô tả ở trên theo định dạng sau đây.

C/C++ programs

```
void Init();
void TypeLetter(char L);
void UndoCommands(int U);
char GetLetter(int P);
```

Pascal programs

```
procedure Init;
procedure TypeLetter(L : Char);
procedure UndoCommands(U : LongInt);
function GetLetter(P : LongInt) : Char;
```

Các chương trình con này phải hoạt động như mô tả ở trên. Tất nhiên bạn được tự do cài đặt thêm các chương trình con khác để sử dụng nội bộ. Các file giao nộp của bạn không được giao tiếp dưới bất cứ hình thức nào với vào/ra chuẩn, cũng như với bất cứ file nào khác.

Sample grader

Chương trình grader sẽ đọc dữ liệu vào theo khuôn dạng:

- dòng 1: số lượng câu lệnh trong input
- trên mỗi dòng tiếp theo:
 - T được tiếp nối bởi dấu cách và chữ cái in thường cho lệnh `TypeLetter`;
 - U được tiếp nối bởi dấu cách và số nguyên cho lệnh `UndoCommands`;
 - P được tiếp nối bởi dấu cách và số nguyên cho lệnh `GetLetter`.

Chương trình grader sẽ in ra các ký tự được đưa ra bởi `GetLetter`, mỗi ký tự trên một dòng riêng biệt.