

## Vėžys perrašinėtojas

Sakoma, kad Leonardo labai žavėjosi vokiečių kalviu Johannes Gutenberg, kuris išrado spausdinimą judančiomis detalėmis, ir, norėdamas jį pagerbti, sukonstravo mašiną, pavadintą vėžiu perrašinėtoju — *il gambero scrivano* — labai paprastą spausdinimo įrenginį, primenantį šiuolaikinius. Įrenginys priima tik dvi komandas: išspausdinti tolesnį simbolį arba anuliuoti paskutiniąsias vykdytas komandas. Vėžio perrašinėtojo atliekama anuliavimo komanda yra ypatinga: anuliavimas taip pat laikomas komanda ir gali būti anuliuotas.

## Užduotis

Realizuokite programinę vėžio perrašinėtojo versiją. Pradiniu momentu teksto nėra (tuščia). Programinis perrašinėtojas priima vartotojo įvestą komandų seką bei užklausas apie tai, koks simbolis yra tam tikroje teksto pozicijoje šiuo metu. Galimos tokios komandos ir užklaustos:

- `Init()` — iškviečiama vieną kartą, be argumentų, pačioje pradžioje. Naudojama inicializuoti duomenų struktūroms. Šios komandos niekada nereikės anuliuoti.
- `TypeLetter(L)` — teksto gale prideda vieną mažąją raidę `L` iš intervalo `a, ..., z`.
- `UndoCommands(U)` — anuliuoja paskutines `U` komandų, kur `U` yra teigiamas sveikasis skaičius.
- `GetLetter(P)` — grąžina raidę, kuri yra teksto pozicijoje `P` užklaustos momentu. Pirmosios teksto raidės pozicija yra 0. Ši užklausa nėra komanda, todėl ji nėra anuliuojama `undo` komanda.

Po pradinio kreipinio į `Init()`, kitos aukščiau aprašytos komandos ir užklaustos gali būti iškviečiamos nulį ar daugiau kartų bet kokia tvarka. Užtikrinama, kad `U` neviršys jau įvykdytų komandų skaičiaus, o `P` bus mažesnis už teksto ilgį (raidžių skaičių tekste) užklaustos metu.

`UndoCommands(U)` anuliuoja `U` komandų *atvirkščia* tvarka: jei anuliuojama komanda yra `TypeLetter(L)`, tuomet pašalinamas paskutinis teksto simbolis (kuris yra `L`); jei anuliuojama komanda yra `UndoCommands(X)`, tuomet pakartotinai įvykdoma `X` ankstesnių komandų jų *pradinė* tvarka.

## Pavyzdys

Parodysime galimą kreipinių seką ir tekstą po kiekvieno kreipinio.

Kreipinys	Gražinama	Tekstas
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

## 1 užduotis [5 taškai]

- Bendras komandų ir užklausų skaičius yra nuo 1 iki 100 (imtinai). Nesikreipiama į UndoCommands.

## 2 užduotis [7 taškai]

- Bendras komandų ir užklausų skaičius yra nuo 1 iki 100 (imtinai). Nebus prašoma anuliuoti UndoCommands.

## 3 užduotis [22 taškai]

- Bendras komandų ir užklausų skaičius yra nuo 1 iki 5 000 (imtinai).

## 4 užduotis [26 taškai]

- Bendras komandų ir užklausų skaičius yra nuo 1 iki 1 000 000 (imtinai). Visos komandos (TypeLetter, UndoCommands) bus pateiktos prieš užklausas GetLetter.

## 5 užduotis [40 taškų]

- Bendras komandų ir užklausų skaičius yra nuo 1 iki 1 000 000 (imtinai).

## Realizacija

Turite pateikti vieną failą, pavadintą `scrivener.c`, `scrivener.cpp` arba `scrivener.pas`. Šiame faile turi būti įrašytos paprogramės su tokiomis antraštėmis.

## Programuojantiems C/C++

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

## Programuojantiems Paskaliu

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

Šios paprogramės turi veikti taip, kaip aprašyta aukščiau. Be abejo, galite sukurti daugiau paprogramių vidiniam vartojimui. Jūsų pateiktas sprendimas neturi dirbti (skaityti ar rašyti) nei su standartiniu įvedimo/išvedimo įrenginiu, nei su jokia kitu failu.

## Pavyzdinis vertintojas

Pavyzdinis vertintojas duomenis perskaito tokiu formtu:

- 1-a eilutė: bendras komandų ir užklausų kiekis duomenyse;
- kiekvienoje tolesnių eilučių:
  - T *mažoji\_lotyniška\_raidė* — nurodo komandą `TypeLetter`;
  - U *sveikasis\_skaičius* — nurodo komandą `UndoCommands`;
  - P *sveikasis\_skaičius* — nurodo komandą `GetLetter`.

Pavyzdinis vertintojas išves procedūros `GetLetter` gražinamus simbolius, kiekvieną atskiroje eilutėje.