



Counting Mushrooms (mushrooms)

Andrew, svjetski priznati ekspert za pečurke, dobio je konačno priliku da proučava pečurke u svom rodnom Singapuru.

U okvir svog istraživanja, Andrew je prikupio n pečurki koje je označio brojevima od 0 do $n - 1$. Svaka od pečurki pripada jednoj od dvije vrste koje nazivamo A i B.

Andrew zna da **pečurka 0 pripada vrsti A**, iako vrste izgledaju slične, on ne zna kojoj vrsti pripadaju ostale pečurke od 1 do $n - 1$.

Na svu sreću, Andrew je dobio uređaj koji mu može pomoći u istraživanju. Uređaj se koristi na sljedeći način: dvije ili više pečurki, u bilo kom redu, mogu se postaviti u uređaj; kada se uključi, uređaj izračunava broj parova **susjednih** pečurki koji su različitih vrsta. Na primjer, ako u uređaj postavite sljedeći niz pečurki $[A, B, B, A]$ (u tom poretku), rezultat će biti 2.

Međutim, upotreba uređaja nije jeftina, pa se može uključiti samo ograničen broj puta. Dodatno, ukupan broj pečurki koje Andrew može staviti u uređaj tokom svih uključivanja uređaja ne smije preći 100 000. Iskoristite ovaj uređaj i pomozite Andrew-u da prebroji koliko ima pečurki vrste A.

Implementacija

Potrebno je implementirati sljedeću funkciju:

```
int count_mushrooms(int n)
```

- n : broj pečurki koje je prikupio Andrew.
- Ova funkcija se poziva tačno jednom i vraća broj pečurki vrste A.

Gornja funkcija može pozvati sljedeću funkciju:

```
int use_machine(int[] x)
```

- x : niz dužine između 2 i n , koji opisuje vrste pečurki postavljenih u uređaj, u tom poretku.
- Elementi niza x moraju biti **različiti** cijeli brojevi od 0 do $n - 1$.
- Neka je d dužina niza x . Tada, funkcija vraća broj različitih indeksa j , takvih da je $0 \leq j \leq d - 2$ i pečurke $x[j]$ i $x[j + 1]$ su različitih vrsta.
- Ovu funkciju možete pozvati najviše 20 000 puta.
- Ukupna dužina svih nizova x koji su predati svim pozivima funkcije `use_machine` ne smije preći 100 000.

Primjer

Primjer 1

Razmotrimo scenario u kojem imamo 3 pečurke čije su vrste redom $[A, B, B]$. Funkciju `count_mushrooms` pozivamo na sljedeći način:

```
count_mushrooms(3)
```

Ova funkcija može pozvati `use_machine([0, 1, 2])`, koja u ovom scenariju vraća 1. Zatim se može pozvati `use_machine([2, 1])`, koja vraća 0.

U ovom trenutku imamo dovoljno informacija da zaključimo da je samo pečurka 1 vrste A, pa će funkcija `count_mushrooms` vratiti 1.

Primjer 2

Razmotrimo scenario u kojem imamo 4 pečurke čije su vrste redom $[A, B, A, A]$. Funkciju `count_mushrooms` pozivamo na sljedeći način:

```
count_mushrooms(4)
```

Ova funkcija može pozvati funkciju `use_machine([0, 2, 1, 3])`, koja vraća 2. Zatim se može pozvati `use_machine([1, 2])`, koja vraća 1.

U ovom trenutku imamo dovoljno informacija da zaključimo da postoje 3 pečurke vrste A. Otuda, funkcija `count_mushrooms` vraća 3.

Ograničenja

- $2 \leq n \leq 20\,000$

Bodovanje

Za sve testove važi da ako poziv funkcije `use_machine` ne zadovoljava navedena pravila ili je vrijednost koju vrati funkcija pogrešna, dobijate 0 bodova,

U suprotnom, neka je Q maksimalan broj poziva funkcije `use_machine` u svim testovima. Bodovi se računaju u skladu sa sljedećom tabelom:

| Uslov | Bodovi |
|----------------------------|---------------------------|
| $20\,000 < Q$ | 0 |
| $10\,010 < Q \leq 20\,000$ | 10 |
| $904 < Q \leq 10\,010$ | 25 |
| $226 < Q \leq 904$ | $\frac{226}{Q} \cdot 100$ |
| $Q \leq 226$ | 100 |

U nekim testovima ponašanje programa za ocjenjivanje (grader-a) je adaptivno. To znači da u tim testovima grader nema fiksirani niz pečurki, već će odgovor zavisiti od prethodnih poziva funkcije `use_machine`. Ipak, garantuje se da će odgovori grader-a biti takvi da pri svakoj interakcij postoji najmanje jedan niz pečurki koji je u skladu sa prethodnim odgovorima.

Primjer programa za testiranje (grader)

Program za testiranje učitava niz cijelih brojeva dužine s koji predstavlja vrste pečuraka. Za sve $0 \leq i \leq n - 1$, $s[i] = 0$ znači da je pečurka i vrste A, dok $s[i] = 1$ znači da je pečurka i vrste B.

Program za testiranje (grader) učitava podatke u sljedećem formatu:

- red 1: n
- red 2: $s[0] \ s[1] \ \dots \ s[n - 1]$

Program za testiranje (grader) štampa rezultat u sljedećem formatu:

- red 1: vrijednost koju vraća `count_mushrooms`.
- red 2: broj poziva `use_machine`.

Obratite pažnju da program za testiranje (grader) iz datog paketa nije adaptivan.