

キノコを数えよう (mushrooms)

キノコ専門家である Andrew はシンガポール原産のキノコを調査している。

調査の一環として、Andrew は 0 から $n - 1$ まで番号づけられた n 個のキノコを収集した。それぞれのキノコは種類 A か種類 B のどちらかである。

Andrew は、キノコ 0 は種類 A であることを知っている。しかし 2 種類のキノコの見た目は同じであるため、彼はキノコ 1 からキノコ $n - 1$ までの各キノコについて、どちらの種類であるかを知らない。

幸運なことに、Andrew の研究室にはこの調査を助ける機械がある。この機械は、2 個以上のキノコを一列に並べて設置して起動すると、キノコの種類が異なるような隣接したキノコのペアの数を報告する。例えば、もしキノコの種類が $[A, B, B, A]$ となるように(この順番で)キノコを設置して起動したら、機械は 2 を報告するだろう。

しかし、この機械を起動するには非常にお金がかかるため、限られた回数しか使うことができない。また、すべての起動におけるキノコの設置個数の合計は 100 000 を超えることができない。この機械を使って、種類 A のキノコの個数を Andrew が数える手助けをしてほしい。

実装の詳細

あなたは以下のプロシーダを実装しなければならない。

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

- n : Andrew が集めたキノコの個数。
- このプロシーダはちょうど 1 回呼び出され、種類 A のキノコの個数を返す必要がある。

`count_mushrooms` プロシーダは、以下のプロシーダを呼び出すことができる。

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

- x : 長さが 2 以上 n 以下の配列で、機械に設置するキノコの番号を順番通りに要素に持っている。
- x の要素は互いに異なる 0 以上 $n - 1$ 以下の整数である必要がある。
- このプロシーダは、 d を x の長さとしたとき、 $0 \leq j \leq d - 2$ についてキノコ $x[j]$ とキノコ $x[j + 1]$ の種類が異なるようなペアの数を返す。
- このプロシーダの呼び出しは 20 000 回以下である必要がある。
- `use_machine` プロシーダのすべての呼び出しにおける x の長さの合計は 100 000 を超えることができない。

入出力例

入出力例1

3 個のキノコがあり、それらの種類が $[A, B, B]$ である場合を考える。count_mushrooms プロシージャは以下のように呼び出される。

```
count_mushrooms(3)
```

count_mushrooms プロシージャが use_machine([0, 1, 2]) を呼び出したとすると(この場合では) 1 が返され、use_machine([2, 1]) を呼び出したとすると 0 が返される。

この時点で、これら 3 個のキノコの中には種類 A のものがちょうど 1 個存在すると結論付けるのに十分な情報があるため、count_mushrooms プロシージャは 1 を返すべきである。

入出力例2

4 個のキノコがあり、それらの種類が $[A, B, A, A]$ である場合を考える。count_mushrooms プロシージャは以下のように呼び出される。

```
count_mushrooms(4)
```

count_mushrooms プロシージャが use_machine([0, 2, 1, 3]) を呼び出したとすると 2 が返され、use_machine([1, 2]) を呼び出したとすると 1 が返される。

この時点で、これら 4 個のキノコの中には種類 A のものがちょうど 3 個存在すると結論付けるのに十分な情報があるため、count_mushrooms プロシージャは 3 を返すべきである。

制約

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

得点計算

もしいくつかのテストケースにおいて、use_machine プロシージャの呼び出しで上に示されたルールが守られていない、もしくは count_mushrooms の戻り値が正しくない場合、その提出に対する得点は 0 となる。そうでない場合、各テストケースにおける use_machine プロシージャの呼び出し回数の最大値を Q とし、以下の表のように得点は計算される。

条件	得点
$20\,000 < Q$	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

いくつかのテストケースにおいては採点プログラムは適応的である。これは、これらのテストケースにおいては採点プログラムは固定されたキノコの種類の列を持たず、それ以前の `use_machine` プロシージャの呼び出しに応じて採点プログラムは応答するということを意味する。ただし、採点プログラムがこのように適応的に応答したとしても、少なくとも 1 つは今までのすべての応答に矛盾しないキノコの種類の列が存在するということが保証される。

採点プログラムのサンプル

採点プログラムのサンプルはキノコの種類を表す整数からなる配列 s を読み込む。

すべての $0 \leq i \leq n - 1$ について、 $s[i] = 0$ はキノコ i が種類 A であることを表し、 $s[i] = 1$ はキノコ i が種類 B であることを表す。

採点プログラムのサンプルの入力形式は以下のとおりである。

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

採点プログラムのサンプルの出力形式は以下のとおりである。

- 1 行目: `count_mushrooms` プロシージャの戻り値。
- 2 行目: `use_machine` プロシージャの呼び出された回数。

注意として、採点プログラムのサンプルは適応的ではない。