FAKULTAS ILMU KOMPUTER

CSGE602040 - Struktur Data dan Algoritma
Semester Ganjil - 2022/2023
Lab 2
Deadline: Jumat, 16 September 2022, 21.00 WIB

Sofita Juga Butuh *Healing*

Deskripsi

Pada suatu hari, Sofita memutuskan untuk *healing* sejenak dengan mencari jajanan yang manis sebagai obat dari rasa jenuh & lelah yang Ia rasakan sebagai anak dari penjual makanan kantin di Fasilkom UI. Sofita memutuskan untuk pergi ke toko jajanan manis yang sangat terkenal di kota tempat tinggalnya, yakni **Cookie Muncher's Dreamland (CMD).** Toko tersebut memiliki suatu keunikan sendiri dalam menjual jajanannya, yakni dengan menggunakan *conveyor belt* yang tidak memiliki ujung. Di atas *conveyor belt* tersebut, terdapat toples-toples yang di dalamnya terdapat kue-kue dengan **4 kemungkinan rasa, yakni Coklat, Stroberi, Vanilla, dan Keju.** *Conveyor belt* tersebut dapat digerakkan dengan menekan tombol panah ke kanan yang akan menggeser *conveyor belt* 1 posisi ke kanan.

Keunikan lainnya dari toko CMD adalah penggunaan toples yang tidak transparan, sehingga Sofita hanya dapat melihat kue paling atas pada toples tersebut. Sebagai anak dari penjual makanan di kantin, Sofita sudah terbiasa melakukan pencatatan yang teliti dan akurat. Oleh karena itu, Sofita sudah menyiapkan strategi yang matang untuk mendapatkan rasa kue yang Ia inginkan, yakni dengan **mencatat posisi setiap toples dengan rasa teratas yang telah lewat persis di depannya.**

Secara sederhana, terdapat dua *query* yang harus diimplementasikan Sofita untuk mendapatkan jajanan yang Ia inginkan, yaitu:

- **GESER_KANAN:** *Query* ini akan menggeser *conveyor belt* ke kanan sehingga mengubah urutan toples 1 posisi ke kanan. Toples yang berada di paling kanan akan berpindah ke paling kiri.
- **BELI_RASA R_i :** Sofita harus menentukan toples mana yang memiliki kue teratas dengan rasa yang Ia cari (rasa R_i). **Untuk membeli kue pada suatu toples, toples tersebut harus berada di posisi terkiri (tempat sofita berdiri).** Sofita dapat memanfaatkan fitur GESER_KANAN pada *conveyor belt* untuk membeli sebuah kue.

Jumlah toples tidak akan berkurang maupun bertambah ketika menjalankan query. Peletakkan toples juga akan dimulai dari kanan ke kiri (toples pertama akan diletakkan di paling kanan conveyor belt dan toples terakhir akan diletakkan di paling kiri conveyor belt).

Bantulah Sofita untuk menemukan jajanan favoritnya secepat & seefisien mungkin agar Ia dapat kembali membantu ibunya berjualan di kantin!

Format Masukan

- Baris pertama terdiri dari satu buah bilangan bulat N yang merupakan banyak toples.
- Baris kedua terdiri dari satu buah bilangan bulat X yang merupakan banyak kue per toples.

- Baris ketiga terdiri dari satu buah bilangan bulat C yang merupakan banyaknya *query* yang akan dijalankan.
- N baris selanjutnya terdiri dari X buah bilangan yang mana X_i berisi rasa kue (R_i) yang akan dimasukkan ke dalam toples ke-N_i.
- C baris selanjutnya terdiri dari string berisi *query* yang akan dijalankan.

Format Keluaran

- **GESER_KANAN**: Sebuah bilangan bulat yang merupakan kode rasa kue paling atas dari toples terkanan **yang telah dipindahkan ke paling kiri.** Jika toples yang dipindahkan kosong, **maka cetak -1.**
- **BELI_RASA:** Sebuah bilangan bulat yang yang merupakan posisi toples yang akan dibeli **(dihitung dari paling kiri, posisi toples mulai dari 0)** sebelum dilakukan pergeseran jika diperlukan**.** Apabila rasa yang ingin dibeli Sofita tidak dijual atau tidak berada di urutan paling atas dari semua toples, serta apabila kue yang Sofita beli merupakan kue terakhir pada toples tersebut, maka **cetak -1.**

Batasan

1.000 ≤ **N** ≤ 25.000
5 ≤ **X** ≤ 200
500 ≤ **C** ≤ 4.000

0 ≤ $\mathbf{R_i}$ ≤ 3, dengan keterangan sebagai berikut:

- 0 kode rasa Coklat
- 1 kode rasa Stroberi
- 2 kode rasa Vanilla
- 3 kode rasa Keju

Contoh Masukan 1

```
4
4
4
0 1 2 3
1 2 3 0
2 3 0 1
3 0 1 2
GESER_KANAN
GESER_KANAN
BELI_RASA 2
BELI_RASA 1
```

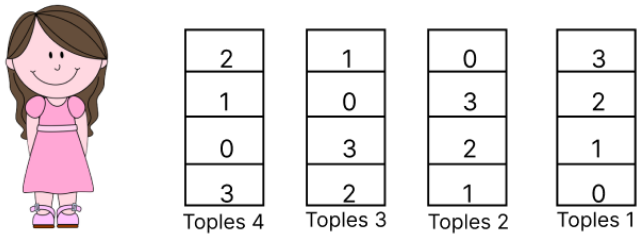
Contoh Keluaran

```
3
0
2
```

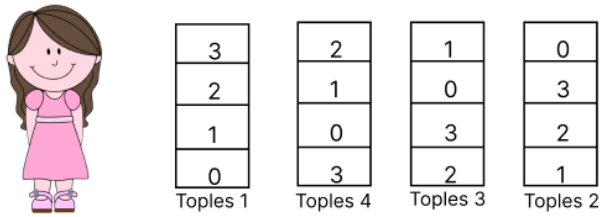
```
0
```

Penjelasan

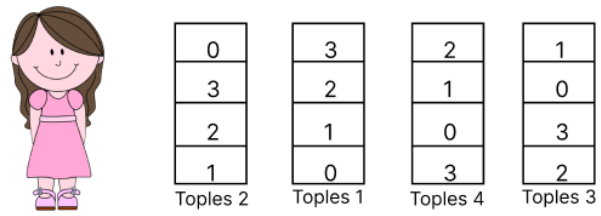
Ilustrasi toples sebelum dilakukan *query.*



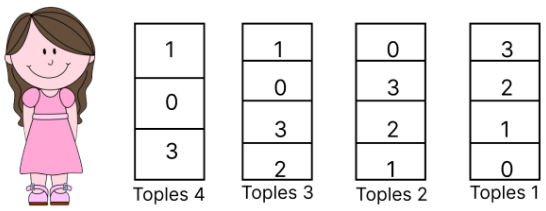
Ilustrasi toples setelah dilakukan query GESER_KANAN pertama.



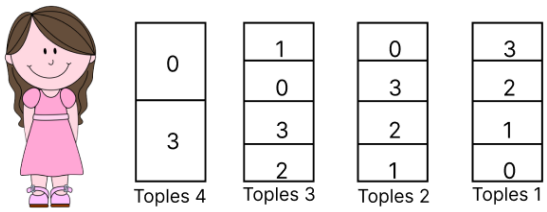
Ilustrasi toples setelah dilakukan query GESER_KANAN kedua.



Ilustrasi toples setelah dilakukan query BELI_RASA 2. Terjadi pergeseran ke kanan sebanyak 2x agar toples yang memiliki kue teratas rasa VANILLA (toples 4) berada di depan Sofita.



Ilustrasi toples setelah dilakukan query BELI_RASA 1. Tidak terjadi pergeseran karena toples yang memiliki kue teratas rasa STROBERI (toples 4) sudah berada di depan Sofita.



Contoh Masukan 2

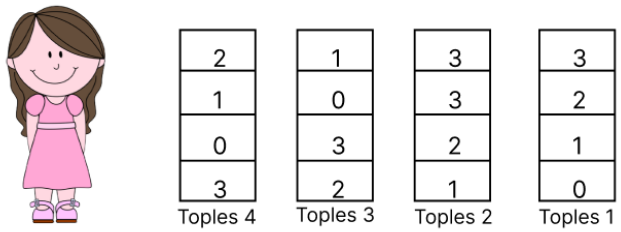
```
4
4
2
0 1 2 3
1 2 3 3
2 3 0 1
3 0 1 2
BELI_RASA 3
BELI_RASA 0
```

Contoh Keluaran

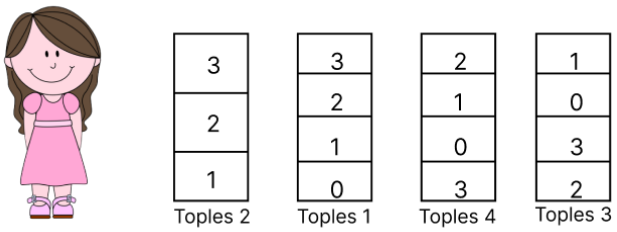
```
2
-1
```

Penjelasan

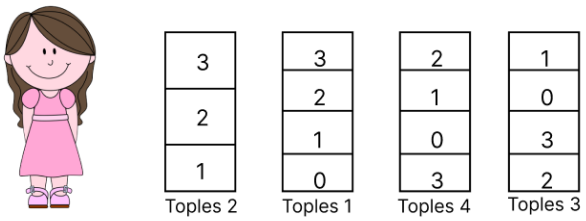
Ilustrasi toples sebelum dilakukan *query.*



Ilustrasi toples setelah dilakukan BELI_RASA 3. Dilakukan pergeseran ke kanan sebanyak 2x agar toples yang memiliki kue teratas rasa KEJU berada di depan Sofita. Perhatikan bahwa toples yang dipilih merupakan toples yang awal mulanya berada paling dekat dengan Sofita.



Ilustrasi toples setelah dilakukan BELI_RASA 0. Kondisi urutan dan isi toples tidak berubah karena kue dengan rasa COKLAT tidak berada di urutan paling atas dari semua toples, sehingga nilai yang dikembalikan adalah -1.



Informasi Tambahan Test-case

Pada 50% test-case berlaku:

- 3.000 <= N <= 7.000