

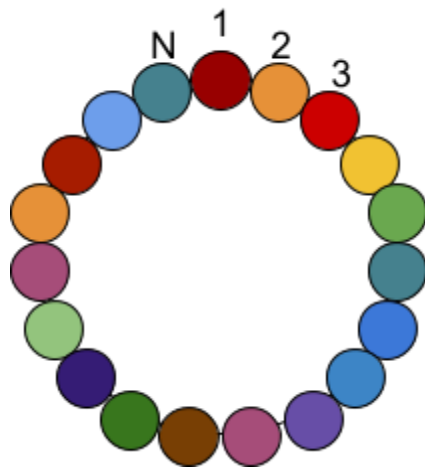
Tugas Pemrograman 2

Kalung

Nama Berkas Kode Sumber : SDA18192T.java
Batas waktu eksekusi program : 1 detik / kasus uji
Batas memori program : 256 MB / kasus uji
Batas pengumpulan : Jumat, 9 November 2018

“Pernahkah anda ingat permainan Zuma Deluxe?”

Dalam suatu permainan, terdapat suatu objek permainan yang digambarkan sebagai rangkaian permata membentuk sebuah kalung. Kalung tersebut terdiri dari sejumlah warna permata. Kita dapat menyisipkan permata baru ke dalam kalung tersebut. Urutan permata dimulai dari yang paling atas, lalu yang berikutnya searah jarum jam, hingga permata ke N yang berada tepat sebelum permata posisi 1. Jika penyisipan terjadi setelah permata posisi K, maka permata tersebut kemudian menjadi permata ke K+1 menggeser permata berikutnya.



Apabila ada deretan permata tak putus minimal 3 buah dengan warna yang sama, maka deretan permata itu akan menghilang dari rangkaian. Demikian pula jika setelahnya terbentuk lagi deretan tak putus minimal 3 buah dengan warna yang sama, deretan itu menghilang kembali, terus menerus sampai tidak ditemukan lagi deretan permata tak putus minimal 3 buah dengan warna yang sama. Karena rangkaian berbentuk kalung, deretan tak putus itu terjadi juga pada permata-permata sekitar posisi N dan 1, yaitu misalnya pada deretan N-C, N-C+1, ..., N, 1, 2, ..., Q. Jika ini terjadi, maka permata pada posisi Q+1 yang kemudian menjadi permata posisi pertama.

Permainan dimulai dengan N buah permata, $4 \leq N \leq 10000$, kemudian selama permainan kalung dapat bertambah panjang, atau bertambah pendek. Permainan berakhir jika rangkaian permata berisi kurang dari 4, atau berisi lebih dari M, $4 \leq N \leq M \leq 18000$.

Dalam versi gamenya aslinya, kalung dibuat berputar setiap saat, dan penyisipan permata baru dilakukan oleh sistem secara acak, atau ditembakkan oleh pemain pada posisi-posisi tertentu sesuai dengan arahan tembakannya. Jika panjang kalung lebih dari M, maka permainan berakhir dengan kekalahan. Jika panjang kalung kurang dari 4, maka permainan berakhir dengan kemenangan.

Instruksi

Kamu tidak diminta membuat game ini *koq* (tapi silahkan kalau kamu tertarik membuatnya, jangan lupa membagi royalty-nya kalau laku dijual buat tim Dosen SDA ya!). Kamu hanya diminta menangani implementasi struktur data untuk menyimpan data permata. Lalu menjalankan event-event yang muncul dari awal hingga akhir yang mengubah susunan permata, sesuai dengan aturan yang dijelaskan di atas.

Anda harus implementasi LinkedList sendiri, tidak diperbolehkan untuk menggunakan library ADT yang terdapat di Java

Masukan

Masukan berisi beberapa baris. Baris pertama berisi N dan M menyatakan banyaknya permata awal dan banyaknya kelereng maksimal sebelum permainan berakhir. Baris kedua berisi oleh N buah bilangan bulat positif yang menyatakan warna-warna permata searah jarum jam.

Pada baris berikutnya terdapat bilangan bulat S, dipastikan $1 \leq S \leq 8000$, yang menyatakan event-event penyisipan permata baru yang akan terjadi. Pada setiap baris dari S baris berikutnya berisi event-event penyisipan permata yang dinyatakan oleh bilangan P dan W, dipisahkan spasi yang menyatakan bahwa posisi penyisipan setelah permata ke P, dan warna permata itu W. Dipastikan $1 \leq P \leq L \leq M$, L adalah panjang rangkaian saat itu dan $1 \leq W \leq 10000$.

Catatan: pada rangkaian awal bisa saja terjadi deretan yang akan menghilang. Jadi sesaat membaca data baris pertama, perlu diperiksa keberadaan deretan tersebut. **Dapat dipastikan setiap waktunya, hanya satu deretan saja yang dihapuskan.**

Keluaran

Pada setiap event, permainan dapat berakhir sebelum semua event diproses. Untuk kondisi ini program mencetak **MENANG** jika jumlah permata kurang dari 4 atau **KALAH** jika jumlah permata lebih dari M dan dalam rangkaian tersebut tidak ada lagi permata yang dapat hilang dari rangkaian. Jika tidak, maka event-event diproses mengubah rangkaian hingga event terakhir dan mencetak suatu angka yang menyatakan banyaknya permata yang tersisa.

Contoh 1

Masukan

10 15

9 8 5 6 2 2 1 1 9 9

4

3 1

3 1

6 2

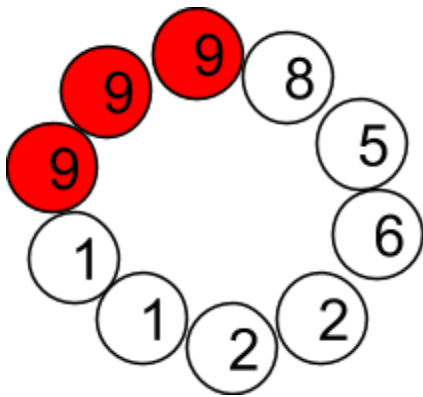
1 5

Keluaran

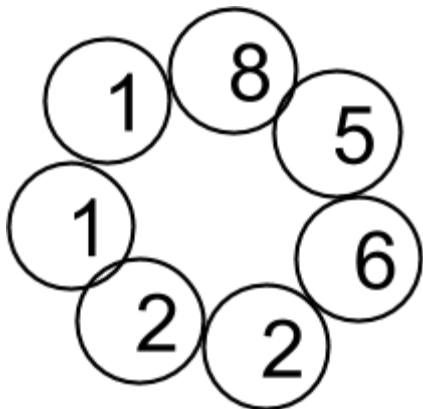
MENANG

Penjelasan:

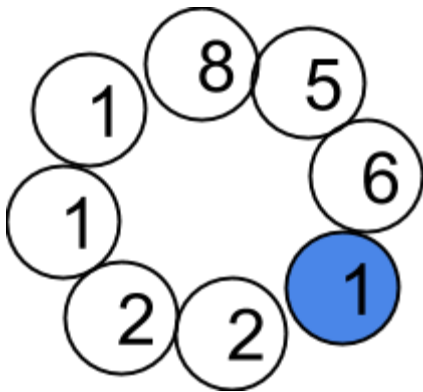
- Rantai awal: 9 8 5 6 2 2 1 1 9 9



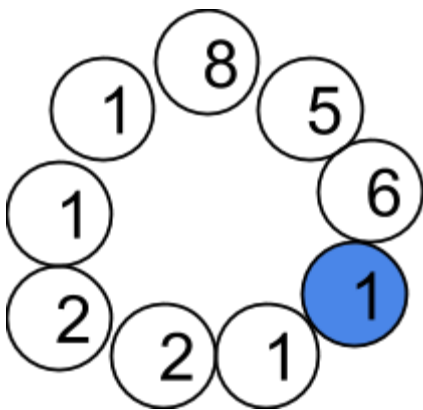
- Setelah menghapus deretan "9 9 9": 8 5 6 2 2 1 1



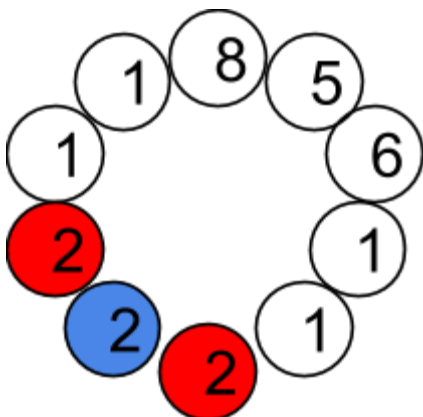
- Setelah event “3 1” pertama: 8 5 6 1 2 2 1 1



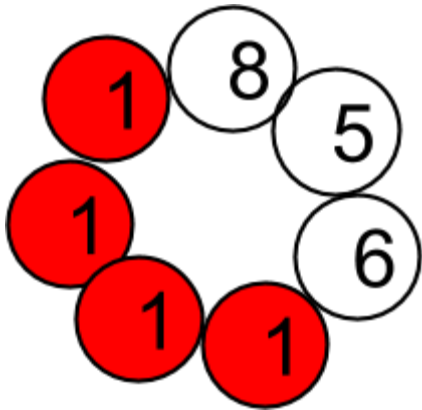
- Setelah event “3 1” kedua: 8 5 6 1 1 2 2 1 1



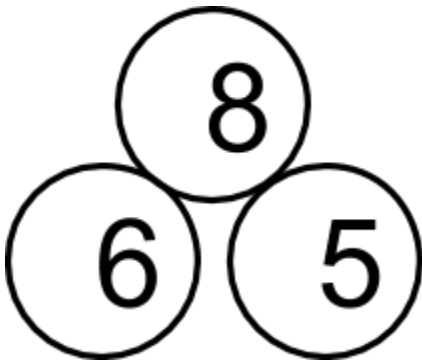
- Setelah event “6 2”: 8 5 6 1 1 2 2 2 1 1



- Setelah menghapus deretan “2 2 2”: 8 5 6 1 1 1 1



- Setelah menghapus deretan “1 1 1 1”: 8 5 6



Permainan berakhir dengan kemenangan. Event setelah itu (“1 5”) diabaikan.

Contoh 2

Masukan

```
5 10
1 2 3 4 5
3
5 1
5 2
5 3
```

Keluaran

```
8
```

Penjelasan:

- Rantai awal: 1 2 3 4 5
- Setelah event “5 1”: 1 2 3 4 5 1
- Setelah event “5 2”: 1 2 3 4 5 2 1
- Setelah event “5 3”: 1 2 3 4 5 3 2 1

Permainan berakhir dan sisa permata berjumlah 8.

Contoh 3**Masukan**

```
6 6
1 2 2 3 4 5
5
2 2
4 1
4 2
4 3
4 4
```

Keluaran

```
KALAH
```

Penjelasan:

- Rantai awal: 1 2 2 3 4 5
- Setelah event “2 2”: 1 2 2 2 3 4 5
- Setelah menghapus deretan “2 2 2”: 1 3 4 5
- Setelah event “4 1”: 1 3 4 5 1
- Setelah event “4 2”: 1 3 4 5 2 1
- Setelah event “4 3”: 1 3 4 5 3 2 1

Permainan berakhir dengan kekalahan, permata berjumlah lebih dari 6 dan tidak ada lagi permata yang dapat menghilang dari rangkaian.