

Hak Cipta Dilindungi  
Undang-undang



## OLIMPIADE SAINS NASIONAL 2015

### DESKRIPSI SOAL



### INFORMATIKA/KOMPUTER

#### SESI – 2

Waktu: 5 Jam

**Daftar Soal:** A. Belanja di Malioboro  
B. Motif Batik  
C. Ayam Aneh

# Belanja di Malioboro

Time limit: 1000 ms

Memory limit: 65536 KB

## Deskripsi

Bebek-bebek Pak Dengklek sangat senang karena Pak Blangkon mengajak mereka untuk pergi berbelanja di Malioboro, suatu tempat berbelanja yang terkenal di Yogyakarta.

Untuk soal ini, Malioboro bisa dianggap sebagai pertokoan melingkar yang terdiri dari  $M$  toko yang diberi nomor dari 1 sampai dengan  $M$ . Jika ditelusuri secara searah jarum jam mulai dari toko nomor 1, toko di sebelahnya adalah toko nomor 2, disusul dengan toko nomor 3, dan seterusnya hingga toko nomor  $M$ , lalu kembali ke toko nomor 1.

Terdapat  $N$  ekor bebek yang pergi ke Malioboro. Bebek ke- $i$  pada awalnya berada di toko nomor  $P_i$ . Kondisi awal ini bisa dianggap terjadi pada menit ke-0. Bebek-bebek sangat bersemangat dalam berbelanja, dan diketahui mereka bergerak dengan kelajuan konstan, yaitu 1 toko per menit. Diketahui bahwa setiap bebek bergerak searah jarum jam atau berlawanan arah jarum jam, dan tidak pernah berganti arah pergerakan.

Pak Blangkon sewaktu-waktu ingin mengetahui jarak terdekat antar bebek. Ia akan bertanya sebanyak  $K$  kali dengan pertanyaan berbunyi "Pada menit ke- $t$ , berapa jarak terdekat antara bebek-bebek yang ada?" Jarak antara dua bebek didefinisikan sebagai banyaknya toko minimal yang perlu dilewati oleh salah satu bebek untuk pergi ke toko tempat bebek lainnya berada.

Bantulah Pak Blangkon menjawab pertanyaannya dengan efisien!

## Format Masukan

Baris pertama akan berisi *label kasus uji*. Label kasus uji adalah sebuah *string* yang dijelaskan sebagai berikut:

- Panjang *string* tersebut adalah banyaknya subsoal ditambah satu.
- Karakter ke-0 (indeks dimulai dari 0) akan berisi 0 jika kasus uji tersebut merupakan contoh kasus uji, atau berisi '.' (titik) jika bukan.
- Untuk setiap nilai  $i$  di antara 1 hingga banyaknya subsoal, berlaku:
  - jika kasus uji tersebut memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi  $i$ , atau
  - jika kasus uji tersebut tidak memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi karakter '.' (titik).

Sebagai contoh apabila label sebuah kasus uji sebuah soal adalah **0..345**, maka:

- Soal tersebut memiliki 5 buah subsoal,
- Kasus uji tersebut merupakan contoh kasus uji, dan
- Kasus uji tersebut memenuhi batasan subsoal ke-3, ke-4, dan ke-5.

Baris berikutnya berisi tiga bilangan bulat, yaitu  $M$ ,  $N$ , dan  $K$ .

N baris berikutnya berisi informasi untuk setiap bebek. Baris ke-i ini berisi dua bilangan, yaitu  $P_i$ ,  $U_i$ . Jika  $U_i$  bernilai -1, artinya bebek ke-i bergerak berlawanan arah jarum jam. Jika  $U_i$  bernilai 1, artinya bebek ke-i bergerak searah jarum jam.

K baris berikutnya berisi sebuah bilangan bulat, yaitu  $t$  yang menyatakan menit ke berapa pada pertanyaan Pak Blangkon. Tidak dijamin bilangan pada baris-baris ini terurut.

### Format Keluaran

Untuk setiap pertanyaan, cetak jarak minimal antara bebek-bebek yang ada pada menit yang ditanyakan.

### Contoh Masukan

```
0 .34567
7 3 4
2 -1
3 1
5 -1
1
2
4
6
```

### Contoh Keluaran

```
0
2
1
1
```

### Penjelasan Contoh

Berikut ini adalah posisi bebek-bebek dari menit 0 sampai menit 6:

t	Bebek 1	Bebek 2	Bebek 3
0	2	3	5
1	1	4	4
2	7	5	3
3	6	6	2
4	5	7	1
5	4	1	7
6	3	2	6

Untuk jawaban pertanyaannya:

- Pada menit ke-1, jarak terdekat adalah 0 (bebek 2 dengan bebek 3).
- Pada menit ke-2, jarak terdekat adalah 2 (bebek 1 dengan bebek 2, atau bebek 2 dengan bebek 3).
- Pada menit ke-4, jarak terdekat adalah 1 (bebek 2 dengan bebek 3).
- Pada menit ke-6, jarak terdekat adalah 1 (bebek 1 dengan bebek 2).

### Subsoal

Untuk setiap subsoal berlaku

- $U_i = -1$ , atau  $U_i = 1$

### Subsoal 1 (5 poin)

Subsoal ini hanya berisi kasus uji berikut ini:

```
.1.34567
30 4 10
5 -1
29 -1
12 1
16 1
6
13
17
7
```

20

1

15

16

2

19

### Subsoal 2 (8 poin)

Subsoal ini hanya berisi kasus uji berikut ini:

..234567

70 7 8

37 1

33 1

59 1

67 -1

10 -1

39 -1

48 -1

5

3

4

12

14

6

11

9

### Subsoal 3 (18 poin)

- $2 \leq N \leq 50$
- $1 \leq t \leq 1.000$
- $1 \leq M \leq 1.000$
- $1 \leq K \leq 1.000$

### Subsoal 4 (27 poin)

- $2 \leq N \leq 1.000$
- $1 \leq t \leq 1.000$
- $1 \leq M \leq 1.000$
- $1 \leq K \leq 1.000$

**Subsoal 5 (14 poin)**

- $2 \leq N \leq 1.000$
- $1 \leq t \leq 250.000$
- $1 \leq M \leq 1.000$
- $1 \leq K \leq 250.000$

**Subsoal 6 (19 poin)**

- $2 \leq N \leq 1.000$
- $1 \leq t \leq 250.000$
- $1 \leq M \leq 250.000$
- $1 \leq K \leq 250.000$

**Subsoal 7 (9 poin)**

- $2 \leq N \leq 1.000$
- $1 \leq t \leq 1.000.000.000 (10^9)$
- $1 \leq M \leq 250.000$
- $1 \leq K \leq 250.000$

# Motif Batik

Time limit: 1000 ms

Memory limit: 65536 KB

## Deskripsi

Yogyakarta memiliki beberapa jenis motif batik. Masing-masing jenis motif memiliki makna dan filosofinya tersendiri.

Pak Dengklek yang sedang berada di Yogyakarta mampir ke sejumlah butik dan membeli  $N$  baju batik. Dari batik-batik yang dibeli Pak Dengklek, terdapat  $M$  jenis motif batik yang dinomori dari 1 sampai dengan  $M$ . Batik ke- $i$  yang dibeli Pak Dengklek memiliki motif jenis  $C_i$ , dan memiliki tingkat kecerahan warna berupa suatu bilangan positif  $W_i$ .

Batik-batik yang dibeli ini akan dipakai oleh keluarga besar Dengklek, dan diabadikan dalam sebuah foto keluarga. Pak Dengklek menyadari bahwa keindahan batik terletak pada keanekaragamannya. Dari  $N$  batik yang telah dibeli, Pak Dengklek ingin mengukur "total keindahan" dari seluruh batiknya. Menurut Pak Blangkon, sang ahli batik, total keindahan dari suatu kumpulan batik adalah jumlah dari selisih tingkat kecerahan warna untuk setiap pasang batik yang berbeda motif.

Sebagai contoh, jika  $N = 5$ ,  $C = [1, 2, 1, 2, 2]$ , dan  $W = [5, 3, 2, 4, 6]$ , maka total keindahannya adalah :

$$\begin{aligned} & |W_1 - W_2| + |W_1 - W_4| + |W_1 - W_5| + |W_2 - W_3| + |W_3 - W_4| + |W_3 - W_5| \\ &= |5 - 3| + |5 - 4| + |5 - 6| + |3 - 2| + |2 - 4| + |2 - 6| \\ &= 2 + 1 + 1 + 1 + 2 + 4 \\ &= 11. \end{aligned}$$

Anda diberikan jenis motif dan tingkat kecerahan warna setiap batik yang dibeli Pak Dengklek. Bantulah Pak Dengklek menentukan total keindahan dari batik-batik tersebut

## Format Masukan

Baris pertama akan berisi *label kasus uji*. Label kasus uji adalah sebuah *string* yang dijelaskan sebagai berikut:

- Panjang *string* tersebut adalah banyaknya subsoal ditambah satu.
- Karakter ke-0 (indeks dimulai dari 0) akan berisi 0 jika kasus uji tersebut merupakan contoh kasus uji, atau berisi '.' (titik) jika bukan.
- Untuk setiap nilai  $i$  di antara 1 hingga banyaknya subsoal, berlaku:
  - jika kasus uji tersebut memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi  $i$ , atau
  - jika kasus uji tersebut tidak memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi karakter '.' (titik).

Sebagai contoh apabila label sebuah kasus uji sebuah soal adalah **0..345**, maka:

- Soal tersebut memiliki 5 buah subsoal,

- Kasus uji tersebut merupakan contoh kasus uji, dan
  - Kasus uji tersebut memenuhi batasan subsoal ke-3, ke-4, dan ke-5.
- Baris kedua berisi dua bilangan bulat, yaitu  $N$  dan  $M$ .

$N$  baris berikutnya berisi dua bilangan bulat. Bilangan-bilangan di baris ke- $i$  ini adalah  $C_i$  dan  $W_i$ .

### Format Keluaran

Sebuah bilangan yang menyatakan total keindahan dari kumpulan batik yang diberikan.

### Contoh Masukan

```
0 . 3456 . 8
5 2
1 5
2 3
1 2
2 4
2 6
```

### Contoh Keluaran

```
11
```

### Subsoal

Pada setiap subsoal, berlaku

- $1 \leq C_i \leq M$
- $1 \leq M \leq N$

### Subsoal 1 (6 poin)

Hanya berisi kasus uji ini:

```
. 1 . 3456 . 8
10 10
10 6
9 10
4 10
2 5
2 7
```



1 10  
3 8  
9 4  
6 7  
6 2

### Subsoal 2 (8 poin)

Hanya berisi kasus uji ini:

..23456.8  
11 3  
3 7  
3 9  
3 10  
1 9  
1 6  
1 10  
1 9  
2 8  
1 1  
3 7  
2 9

### Subsoal 3 (34 poin)

- $1 \leq N \leq 1.000$
- $1 \leq M \leq 50$
- $1 \leq W_i \leq 200$

### Subsoal 4 (6 poin)

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 50$
- $1 \leq W_i \leq 200$

### Subsoal 5 (7 poin)

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 200$
- $1 \leq W_i \leq 200$

**Subsoal 6 (9 poin)**

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 2.000$
- $1 \leq W_i \leq 2.000$

**Subsoal 7 (13 poin)**

- $1 \leq N \leq 10^5$
- $M = N$
- $C_i = i$
- $1 \leq W_i \leq 10^9$

**Subsoal 8 (17 poin)**

- $1 \leq N \leq 10^5$
- $1 \leq W_i \leq 10^9$

**Peringatan**

Bagi pengguna C/C++, gunakan `"%lld"` atau `cin/cout` untuk membaca/menulis bilangan bulat 64-bit.

# Ayam Aneh

Time limit: 100 ms

Memory limit: 32768 KB

## Deskripsi

Untuk menghemat biaya perjalanan, Pak Dengklek dan bebek-bebeknya menginap di rumah Pak Blangkon. Di sana, Pak Dengklek melihat kebun Pak Blangkon yang penuh dengan ayam. Warna ayam Pak Blangkon bermacam-macam. Setiap hari, Pak Blangkon memberikan ayamnya makanan. Makanan ini dapat mengakibatkan DNA seekor ayam bermutasi dan alhasil menjadikan warna ayam tersebut berubah.

Pak Dengklek yang tertarik melihat warna ayam-ayam itu ingin mencoba mengubah warna bebek-bebeknya sendiri. Pak Dengklek pun bertanya kepada Pak Blangkon tentang DNA ayam-ayam yang dimiliki Pak Blangkon agar Pak Dengklek bisa mengimplementasikannya ke bebeknya sendiri. Tetapi Pak Blangkon yang iseng menyuruh Pak Dengklek menebak sendiri apa DNA ayamnya tersebut.

DNA ayam-ayam tersebut dapat direpresentasikan sebagai sebuah string yang terdiri atas N huruf kapital dari A sampai dengan Z. **Semua huruf yang sama dalam sebuah string DNA pasti akan berada bersebelahan.** Sebagai contoh, AAASSSDDDDFF adalah sebuah string DNA, sedangkan AAASSSDDDAFFFF bukan karena di antara A terdapat kelompok huruf S dan D. Cara Pak Dengklek menebak adalah sebagai berikut: Pak Blangkon memberikan suatu bilangan N yang merupakan panjang string DNA dari ayam yang ingin Pak Dengklek ketahui DNA-nya. Selain itu, Pak Blangkon juga memberikan bilangan K yang merupakan batas pertanyaan yang boleh Pak Dengklek tanyakan sebelum Pak Dengklek harus menebak string DNA yang dimiliki ayam tersebut.

Setiap kali Pak Dengklek bertanya, ia akan memberikan suatu string. Setelah itu Pak Blangkon akan membalas YA atau TIDAK berdasarkan apakah string yang ditanyakan Pak Dengklek merupakan substring dari DNA ayam yang sedang ingin Pak Dengklek ketahui (YA jika substring dan TIDAK jika bukan). Sebagai informasi, X merupakan substring dari Y apabila X terdiri atas setidaknya satu karakter, dan X dapat dihasilkan dengan membuang nol atau lebih huruf-huruf awalan Y dan nol atau lebih huruf-huruf akhiran Y. Sebagai contoh, BEBE, EBE, BEK, dan BEBEK adalah substring dari BEBEK, sedangkan BEEK bukan substring dari BEBEK.

Pak Dengklek pun kebingungan dengan cara untuk menebak DNA ayam-ayam Pak Blangkon. Bantulah Pak Dengklek menentukan DNA ayam Pak Blangkon!

## Format Interaksi

Pada mulanya, program Anda akan menerima sebuah baris berisi *label kasus uji*. Label kasus uji adalah sebuah *string* yang dijelaskan sebagai berikut:

- Panjang *string* tersebut adalah banyaknya subsoal ditambah satu.
- Karakter ke-0 (indeks dimulai dari 0) akan berisi 0 jika kasus uji tersebut merupakan contoh kasus uji, atau berisi '.' (titik) jika bukan.

- Untuk setiap nilai  $i$  di antara 1 hingga banyaknya subsoal, berlaku:
  - jika kasus uji tersebut memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi  $i$ , atau
  - jika kasus uji tersebut tidak memenuhi batasan subsoal ke- $i$ , maka karakter ke- $i$  berisi karakter '.' (titik).

Sebagai contoh apabila label sebuah kasus uji sebuah soal adalah **0.345**, maka:

- Soal tersebut memiliki 5 buah subsoal,
- Kasus uji tersebut merupakan contoh kasus uji, dan
- Kasus uji tersebut memenuhi batasan subsoal ke-3, ke-4, dan ke-5.

Kemudian, program Anda akan menerima sebuah baris berisi dua buah bilangan bulat  $N$  dan  $K$ . Setelah itu, program Anda dapat melakukan serangkaian tindakan, yang masing-masing merupakan salah satu dari:

- Bertanya. Program Anda harus mengeluarkan sebuah baris berisi **TANYA** `<pertanyaan>` yakni, string **TANYA** diikuti dengan substring yang Anda tanyakan. Setiap kali program Anda selesai bertanya, program Anda membaca sebuah string yang dijamin merupakan salah satu dari **YA** atau **TIDAK**, sesuai dengan definisi pada deskripsi soal.
- Menebak. Program Anda harus mengeluarkan sebuah baris berisi **JAWAB** `<jawaban>` yakni, string **JAWAB** diikuti dengan string DNA tebakan Anda. Program Anda **harus berhenti** setelah melakukan ini.

### Contoh Interaksi

Misalkan string DNA yang sebenarnya adalah **BAASSDFFF**. Berikut adalah contoh interaksi yang mungkin terjadi.

Keluaran Program Anda	Keluaran Program Grader
	0..... 10 3
TANYA ABCD	
	TIDAK
TANYA AASSDFFF	
	YA
TANYA DFFF	

Keluaran Program Anda	Keluaran Program Grader
	YA
JAWAB BAASSDDFFF	
	(interaksi selesai)

### Subsoal

Pada semua subsoal, berlaku:

- String DNA hanya terdiri atas huruf kapital A s.d. Z

#### Subsoal 1 (6 poin)

- $N = 10$
- $K = 10$
- String DNA hanya terdiri atas huruf A, B, atau C

#### Subsoal 2 (10 poin)

- $N = 10$
- $K = 350$

Khusus untuk subsoal 1 dan subsoal 2:

- Hanya terdapat sebuah kasus uji (satu subsoal dinyatakan oleh satu kasus uji), yang berupa game dan dapat dilihat di halaman pengumuman kontes.
- Anda boleh memainkan permainan ini berulang kali tanpa mendapatkan penalti.
- Jika Anda sudah memenangkan permainan untuk subsoal tertentu, Anda dapat memilih pilihan pada permainan untuk mengeluarkan *source code* yang dapat langsung Anda kirimkan ke *grader* dan menjawab dengan benar pada subsoal yang telah Anda menangkan.
- Anda tidak diwajibkan memainkan permainan ini untuk mengerjakan kedua subsoal ini. Anda diperbolehkan untuk menulis kode Anda sendiri untuk mengerjakan kedua subsoal ini.

### Subsoal 3 (31 poin)

- $N = 26$
- $K = 350$
- String DNA mengandung semua karakter dari A s.d. Z masing-masing tepat sekali

### Subsoal 4 (24 poin)

- $1 \leq N \leq 100$
- $K = 500$

### Subsoal 5 (29 poin)

- $1 \leq N \leq 1.000$
- $K = 750$

### Catatan

Yang perlu diperhatikan adalah bahwa untuk tipe soal interaktif seperti ini, Anda harus selalu memberikan perintah `fflush(stdout)`; (bagi pengguna C/C++) atau `flush(output)`; (bagi pengguna PASCAL) setiap kali Anda mencetak keluaran (dengan kata lain, setiap kali ada perintah mencetak keluaran misalnya `write`, `writeln`, `printf`, `cout`, atau `puts`, tepat di bawahnya harus ada perintah `fflush/flush`).

Sebagai contoh, berikut adalah contoh *source code* dalam bahasa Pascal yang akan selalu bertanya "TANYA ABC" dan kemudian menjawab "JAWAB DEF" tanpa mempedulikan nilai N dan K yang diberikan maupun bacaan hasil interaksi.

```
var subsoal, jawaban: string;
    N, K: longint;

begin
    readln(subsoal);
    readln(N, K);

    writeln('TANYA ABC');
    flush(output);

    readln(jawaban);

    writeln('JAWAB DEF');
    flush(output);
end.
```

Dan berikut adalah contoh *source code* yang melakukan hal yang sama dalam bahasa C++.

```
#include <cstdio>
```

```

#include <cstring>
char subsoal[100], jawaban[100];
int N, K;

int main() {
    gets(subsoal);
    scanf("%d %d", &N, &K);

    printf("TANYA ABC\n");
    fflush(stdout);

    gets(jawaban);

    printf("JAWAB DEF\n");
    fflush(stdout);

    return 0;
}

```

### Peringatan

Jika program Anda melakukan salah satu dari hal-hal di bawah ini:

- melakukan tindakan di luar format dan batasan yang ditentukan,
- bertanya lebih dari K kali, atau
- salah menjawab DNA

maka nilai Anda untuk subsoal yang bersangkutan adalah nol.

Selain itu, pastikan pula bahwa program Anda harus berhenti jika selesai melakukan interaksi. Bila tidak, maka Anda mungkin mendapatkan *Time Limit Exceeded*, *Runtime Error*, atau *Internal Error* untuk kasus uji yang bersangkutan.