



**SOAL UJIAN  
OLIMPIADE SAINS NASIONAL 2013  
CALON PESERTA  
INTERNATIONAL OLYMPIAD IN INFORMATICS (IOI) 2014  
HARI KE-1**



**INFORMATIKA**

Waktu : 5 jam

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN  
DIREKTORAT JENDERAL PENDIDIKAN MENENGAH  
DIREKTORAT PEMBINAAN SEKOLAH MENENGAH ATAS  
TAHUN 2013**

# Tebak Himpunan

Batas Waktu 1 detik

Batas Memori 32 MB

## Deskripsi

Pak Dengklek kini sudah pensiun dan memiliki banyak cucu. Untuk mengisi waktu luangnya, Pak Dengklek sering membuat permainan yang bisa dimainkan saat cucu-cucunya datang.

Saat liburan sekolah, cucu kesayangan Pak Dengklek berlibur ke rumahnya. Ternyata, ia telah menyiapkan permainan baru untuk cucu kesayangannya tersebut! Permainannya adalah sebagai berikut.

Pak Dengklek mengumumkan sebuah bilangan bulat positif  $N$ . Setelah itu, ia memilih subhimpunan  $S$  dari  $\{1, 2, \dots, N\}$ . Tugas sang cucu adalah menebak  $S$  dengan paling banyak  $Q$  buah tebakan kepada Pak Dengklek. Setiap tebakan berupa sebuah subhimpunan  $T$  dari  $\{1, 2, \dots, N\}$  yang harus memiliki setidaknya  $K$  anggota.

Untuk setiap tebakan, Pak Dengklek akan menjawab salah satu dari ketiga jawaban berikut:

- YA, apabila  $S$  sama persis dengan  $T$ .
- BISA JADI, apabila  $S$  tidak sama dengan  $T$  namun ada minimal satu anggota  $S$  yang juga merupakan anggota  $T$ .
- TIDAK, apabila tidak ada satupun anggota  $S$  yang juga merupakan anggota  $T$ .

Sebagai contoh, apabila  $N = 10$ ,  $S = \{2, 3, 5, 7\}$ , dan  $K = 2$ :

- Jika  $T = \{2, 3, 5, 7\}$ , maka Pak Dengklek menjawab YA.
- Jika  $T = \{2, 3, 7\}$ , maka Pak Dengklek menjawab BISA JADI.
- Jika  $T = \{1, 2, 3, 5, 7\}$ , maka Pak Dengklek menjawab BISA JADI.
- Jika  $T = \{2, 8\}$ , maka Pak Dengklek menjawab BISA JADI.
- Jika  $T = \{1, 8, 9\}$ , maka Pak Dengklek menjawab TIDAK.
- Jika  $T = \{4, 10\}$ , maka Pak Dengklek menjawab TIDAK.
- $T = \{5\}$  tidak diperbolehkan karena banyak anggotanya kurang dari  $K$ .

Sekarang, Anda diminta membuat sebuah program interaktif yang dapat membantu cucu Pak Dengklek memenangkan permainan ini.

## Format Interaksi

Awalnya, program Anda harus membaca sebuah baris berisi string "Subsoal  $X$ ", dengan  $X$  adalah nomor subsoal. Kemudian, program Anda harus membaca tiga buah bilangan bulat  $N$ ,  $K$ , dan  $Q$ , dipisahkan oleh spasi. Setelah itu, program Anda dapat mengeluarkan paling banyak  $Q$  buah tebakan dalam format:

$M \ T_1 \ T_2 \ T_3 \ \dots \ T_M$

yakni, sebuah bilangan bulat  $M$  diikuti dengan  $M$  buah bilangan bulat dipisahkan spasi, yang berarti Anda memberi tebakan  $T = \{T_1, T_2, \dots, T_M\}$ , dengan syarat:

- $K \leq M \leq N$
- $T_1 < T_2 < \dots < T_M$

Setiap kali program Anda selesai mengeluarkan tebakan, program Anda membaca sebuah

string yang mendeskripsikan jawaban Pak Dengklek. String tersebut dijamin selalu merupakan salah satu dari:

- "ya", artinya Pak Dengklek menjawab YA. Anda langsung dianggap benar dalam kasus uji yang bersangkutan dan program Anda tidak perlu melakukan apa-apa lagi.
- "bisajadi", artinya Pak Dengklek menjawab BISA JADI.
- "tidak", artinya Pak Dengklek menjawab TIDAK.

Pastikan program Anda berhenti melakukan interaksi setelah menerima jawaban "ya".

### Contoh Interaksi

Berikut adalah contoh interaksi program, dengan subhimpunan **S** yang dimiliki *grader* adalah  $S = \{2, 5, 6\}$ .

Keluaran Program Peserta	Umpan Balik <i>Grader</i>
	Subsoal 0
	7 3 10
3 1 2 6	
	bisajadi
4 1 3 4 7	
	tidak
7 1 2 3 4 5 6 7	
	bisajadi
3 2 5 7	
	bisajadi
3 2 5 6	
	ya
(interaksi selesai)	

### Pembagian Subsoal

Pada semua subsoal, berlaku:

- **S** memiliki setidaknya **K** buah anggota.

#### Subsoal 1 (6 poin)

- **N** = 5
- **K** = 1
- **Q** = 32

#### Subsoal 2 (12 poin)

- **N** = 6

- $K = 2$
- $Q = 100$

Khusus untuk subsoal 1 dan subsoal 2:

- Hanya terdapat sebuah kasus uji (satu subsoal dinyatakan oleh satu kasus uji), yang dapat dimainkan [di sini](#).
- Dalam permainan tersebut, banyaknya tebakan yang dapat diajukan tidak dibatasi.
- 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.

### Subsoal 3 (5 poin)

- $1 \leq N \leq 10$
- $K = 1$
- $Q = 2^N$

### Subsoal 4 (8 poin)

- $1 \leq N \leq 100$
- $K = 1$
- $Q = N + 1$

### Subsoal 5 (33 poin)

- $1 \leq N \leq 1.000$
- $K = 1$
- $Q = 2 \times \text{ceil}(\log_2 N) + 1$
- $S$  selalu berupa  $\{1, 2, \dots, R\}$  dengan  $R \leq N$

### Subsoal 6 (36 poin)

- $2 \leq N \leq 100$
- $K = 2$
- $Q = N^2$

### Catatan

- $\text{ceil}(x)$  adalah bilangan bulat terkecil yang lebih besar dari atau sama dengan  $x$ .

- Sebuah himpunan  $V$  dikatakan merupakan subhimpunan dari himpunan  $S$  apabila setiap anggota  $V$  juga merupakan anggota  $S$ .

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 menebak semua  $N$  barang tanpa mempedulikan nilai  $N$ ,  $K$ , dan  $Q$  yang diberikan.

```
var subsoal: string;
    N, K, Q, i: longint;

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

    write(N);
    for i := 1 to N do write(' ', i);
    writeln;
    flush(output);
end.
```

Dan berikut adalah contoh *source code* dalam bahasa C++.

```
#include <cstdio>
#include <cstring>

char subsoal[100];
int nomor;
int N, K, Q, i;

int main() {
    scanf("%s %d", subsoal, &nomor);
    scanf("%d %d %d", &N, &K, &Q);

    printf("%d", N);
```

```
for (i = 1; i <= N; i++) printf(" %d", i);  
printf("%n");  
fflush(stdout);  
  
return 0;  
}
```

## Peringatan

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

- mengeluarkan tebakan atau menjawab tidak sesuai format sehingga tidak dikenali oleh *grader*, atau
- menebak lebih dari **Q** kali,

maka program Anda akan dihentikan secara otomatis dan Anda tidak memperoleh nilai pada kasus uji yang bersangkutan.

Perhatikan bahwa untuk soal ini, jika solusi Anda dapat menyelesaikan subsoal **X**, tidak dijamin bahwa solusi tersebut juga dapat menyelesaikan subsoal-subsoal **Y** dengan  $Y < X$ .

# Berbaris Sebelum Masuk

Batas Waktu 1 detik

Batas Memori 32 MB

## Deskripsi

Pak Dengklek adalah seorang wali murid di SD TOKI. Setiap pagi, para siswa diharuskan untuk berbaris di depan pintu kelas sebelum masuk. Namun, Pak Dengklek memiliki sedikit kesulitan karena tiap siswa memiliki keinginan tersendiri dalam berbaris.

Terdapat  $N$  orang siswa di SD tersebut, dinomori dari 1 sampai dengan  $N$ . Siswa nomor 1 berada di paling depan dan siswa nomor  $N$  berada di paling belakang barisan. Siswa ke- $i$  memiliki tinggi badan sebesar  $T_i$  satuan. Setiap siswa ke- $i$  ingin agar saat ia berbaris, banyaknya siswa di depannya yang memiliki tinggi badan kurang dari atau sama dengan tinggi badannya, berada di antara  $A_i$  dan  $B_i$  orang siswa, inklusif.

Bantulah Pak Dengklek membariskan siswa-siswanya dalam satu baris sedemikian sehingga semua keinginan siswanya terpenuhi.

## Format Masukan

Baris pertama pada berkas masukan berisi string "Subsoal  $X$ " dengan  $X$  menyatakan nomor subsoal.

Baris kedua berisi sebuah bilangan bulat  $N$ .  $N$  baris berikutnya masing-masing berisi 3 buah bilangan bulat  $T_i$ ,  $A_i$ , dan  $B_i$ .

## Format Keluaran

Keluarkan salah satu barisan yang memenuhi, dalam format

$$S_1 S_2 S_3 \dots S_N$$

dengan  $S_i$  adalah nomor siswa yang berada pada posisi ke- $i$  dari depan.

## Contoh Masukan

Subsoal 0

3

150 1 3

160 2 2

140 0 2

## Contoh Keluaran

3 1 2

## Pembagian Subsoal

Pada semua subsoal, berlaku:

- $1 \leq T_i \leq 1.000$

- $0 \leq A_i \leq B_i \leq N$
- Semua nilai  $T_i$  berbeda-beda
- Dijamin ada setidaknya sebuah barisan yang memenuhi syarat

**Subsoal 1 (9 poin)**

- Hanya terdapat sebuah kasus uji, yang dapat diunduh [di sini](#).

**Subsoal 2 (20 poin)**

- Hanya terdapat sebuah kasus uji, yang dapat diunduh [di sini](#).

**Subsoal 3 (21 poin)**

- $1 \leq N \leq 8$

**Subsoal 4 (50 poin)**

- $1 \leq N \leq 1.000$



# Lipat Kertas

Batas Waktu 0,2 detik

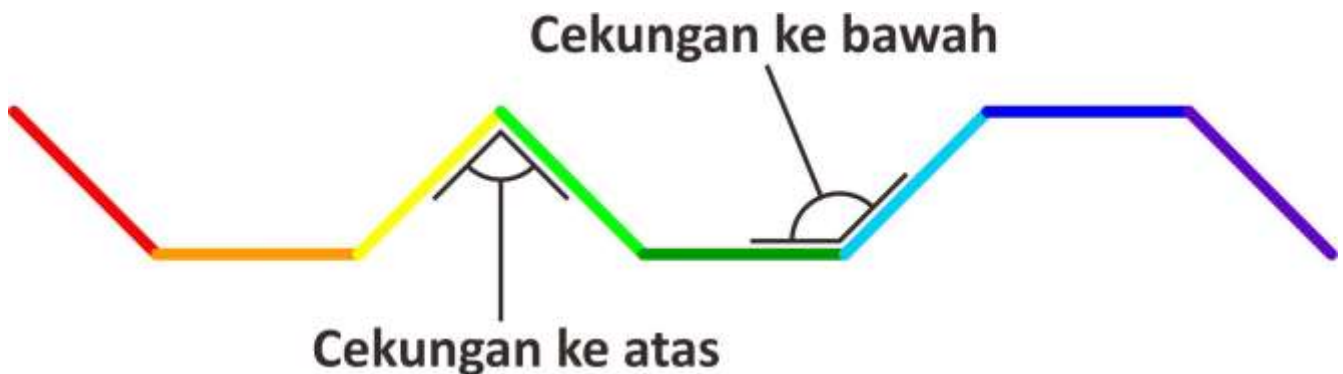
Batas Memori 32 MB

## Deskripsi

Pak Dengklek sekarang sedang sibuk memainkan kertas origami buatan Pak Ganesh yang berbentuk persegi panjang dengan ukuran  $1 \times N$  petak berwarna-warni. Petak warna paling kiri disebut petak 1, di kanannya disebut petak 2, dan seterusnya. Kertas origami ini unik, karena bagian yang dapat dilipat hanyalah sisi persinggungan 2 buah petak saja. Dengan demikian, terdapat  $N - 1$  lekukan yang dapat dilipat.



Pak Ganesh sebelumnya telah melipat-lipat kertasnya menjadi berukuran  $1 \times 1$  (tentu hasilnya sangatlah tebal!). Kemudian, Pak Ganesh membuka lipatan-lipatan kertas tersebut. Hasilnya, kertas tersebut memiliki bekas lekukan-lekukan pada bagian yang dilipat dan lekukan-lekukan tersebut menjadi cekungan ke atas ataupun cekungan ke bawah.



Gambar di atas merupakan kondisi kertas yang dibuka dari lipatannya dan dilihat dari sisi samping ketika permukaan kertas menghadap ke atas dan petak nomor 1 berada di paling kiri.

Kali ini Pak Ganesh tidak lagi memberikan Pak Dengklek kertas yang dibuka dari lipatannya, melainkan Pak Ganesh memberikan urutan petak warna hasil akhir lipatan apabila dilihat dari samping kertas ketika permukaan kertas menghadap ke atas dan petak nomor 1 berada di paling kiri. Pak Dengklek diminta untuk mencari tahu bagaimana kondisi lekukan kertas tersebut apabila dibuka dari lipatannya!

## Format Masukan

Baris pertama pada berkas masukan berisi string "Subsoal X" (tanpa tanda kutip) dengan X menyatakan nomor subsoal. Baris kedua berisi sebuah bilangan bulat N, yaitu banyaknya petak warna-warni pada kertas. Baris ketiga berisi N buah bilangan bulat dipisahkan spasi, yaitu urutan petak warna hasil akhir lipatan apabila dilihat dari sisi samping kertas ketika permukaan kertas menghadap ke atas dan petak nomor 1 berada di paling kiri..

## Format Keluaran

Keluaran terdiri dari 1 baris yang berisi string S.

String S memiliki panjang N – 1 karakter yang menunjukkan kondisi kertas setelah dibuka dari lipatannya. Karakter ke-i pada S menunjukkan lekukan di antara petak i dan petak i + 1. Karakter 'A' menunjukkan lekukan tersebut berupa cekungan ke atas, dan karakter 'B' menunjukkan lekukan tersebut berupa cekungan ke bawah.

Apabila tidak ada kemungkinan solusi, S berisi string "INVALID" (tanpa tanda kutip). Apabila terdapat lebih dari satu kemungkinan solusi, keluarkan yang mana saja.

### Contoh Masukan 1

```
Subsoal 0  
8  
1 8 5 4 3 6 7 2
```

### Contoh Keluaran 1

```
BBABBAA
```

### Contoh Masukan 2

```
Subsoal 0  
4  
4 1 3 2
```

### Contoh Keluaran 2

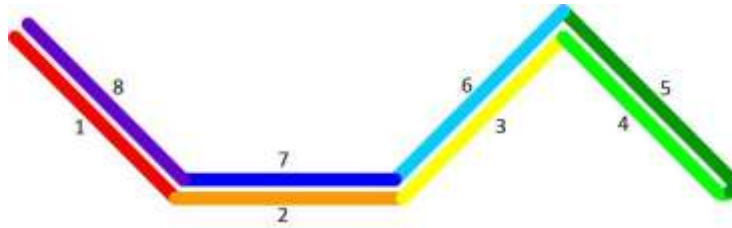
```
INVALID
```

## Penjelasan Contoh Kasus Uji



Gambar berikut adalah kondisi kertas yang ditunjukkan oleh string S pada contoh keluaran 1.

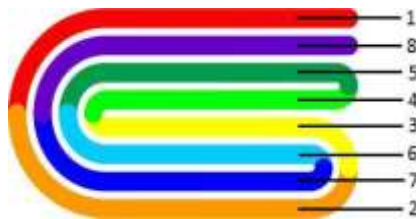
Kemudian cobalah lipat pada bagian di antara petak 4 dan petak 5.



Selanjutnya, lipatlah bagian antara petak 2 dan petak 3 maupun bagian antara petak 7 dan petak 6.



Terakhir, lipatlah lipatan bagian yang tersisa.



Didapat hasil akhir lipatan dengan urutan petak warna : 1 8 5 4 3 6 7 2 (dilihat dari bagian atas ke bawah), yang merupakan urutan petak warna yang ditanyakan oleh Pak Ganesh pada contoh masukan 1.

Perhatikan bahwa apabila kertas tersebut dibuka kembali, maka kondisi lekukan kertas tidak berbeda dengan kondisi lekukan kertas sebelum dilipat. Sehingga, lekukan kertas tersebut merupakan salah satu solusi yang mungkin.

## Pembagian Subsoal

### Subsoal 1 (9 poin)

- Hanya terdapat sebuah kasus uji, yang dapat diunduh [di sini](#).

### Subsoal 2 (9 poin)

- Hanya terdapat sebuah kasus uji, yang dapat diunduh [di sini](#).

**Subsoal 3 (30 poin)**

- $2 \leq N \leq 1.000$
- Tidak ada masukan dengan keluaran INVALID

**Subsoal 4 (7 poin)**

- $2 \leq N \leq 4$

**Subsoal 5 (28 poin)**

- $2 \leq N \leq 1.000$

**Subsoal 6 (17 poin)**

- $2 \leq N \leq 100.000$

# Mengelindingkan Kubus

Batas Waktu 1 detik

Batas Memori 32 MB

## Deskripsi

Diberikan sebuah kubus yang masing-masing muka sisinya dinomori antara 1 hingga 6, inklusif. Kubus itu diletakkan pada bidang datar yang luasnya tidak terbatas.

Pada mulanya, masing-masing sisi kubus itu menghadap ke arah atas, bawah, utara, timur, selatan, dan barat yang secara berturut-turut dinomori dengan  $P_1$ ,  $P_2$ ,  $P_3$ ,  $P_4$ ,  $P_5$ , dan  $P_6$ .

Anda diperbolehkan menggelindingkan kubus itu ke arah utara, timur, barat, atau selatan.

Akibat dari menggelindingkan kubus sekali adalah perubahan konfigurasi nomor pada muka sisinya yang tergantung pada arah kubus digelindingkan.

Berikut ini adalah contoh sebuah kubus yang kebetulan dibuat mirip dengan dadu. Setiap langkah dilakukan sekali dan merupakan lanjutan dari langkah sebelumnya.

Muka Kubus	Awal	Digelindingkan ke Utara	Kemudian ke Timur	Kemudian ke Selatan	Kemudian ke Barat
$P_1$ (atas)	1	5	3	1	5
$P_2$ (bawah)	6	2	4	6	2
$P_3$ (utara)	2	1	1	4	4
$P_4$ (timur)	4	4	5	5	6
$P_5$ (selatan)	5	6	6	3	3
$P_6$ (barat)	3	3	2	2	1

Tugas Anda adalah menggelindingkan kubus dengan sesedikit mungkin langkah sedemikian sehingga konfigurasi nomor-nomor pada muka sisi atas, bawah, utara, timur, selatan, dan barat secara berturut-turut menjadi  $Q_1$ ,  $Q_2$ ,  $Q_3$ ,  $Q_4$ ,  $Q_5$ , dan  $Q_6$ .

## Format Masukan

Baris pertama pada berkas masukan berisi string "Subsoal X" dengan X menyatakan nomor subsoal. Baris kedua berisi 6 buah bilangan bulat dipisahkan spasi yang secara berturut-turut menyatakan  $P_1$ ,  $P_2$ ,  $P_3$ ,  $P_4$ ,  $P_5$ , dan  $P_6$ . Baris ketiga berisi 6 buah bilangan bulat dipisahkan spasi yang secara berturut-turut menyatakan  $Q_1$ ,  $Q_2$ ,  $Q_3$ ,  $Q_4$ ,  $Q_5$ , dan  $Q_6$ .

## Format Keluaran

Sebuah baris berisi sebuah bilangan bulat yang menyatakan banyaknya langkah penggelindingan yang diperlukan agar tujuan seperti pada deskripsi soal tercapai dan banyaknya langkah penggelindingan tersebut minimal.

## Contoh Masukan

```
Subsoal 0
1 4 3 2 3 4
```

3 3 4 4 2 1

## Contoh Keluaran

2

## Penjelasan Contoh Kasus Uji

Pada contoh masukan, salah satu cara untuk mendapatkan konfigurasi akhir dalam dua langkah adalah gelindingkan kubus ke barat, kemudian ke selatan. Setelah langkah pertama, konfigurasi kubus adalah 2 4 3 4 3 1.

## Pembagian Subsoal

Pada semua subsoal, berlaku:

- Nilai-nilai  $Q_1, Q_2, Q_3, Q_4, Q_5$ , dan  $Q_6$  adalah sedemikian sehingga terdapat cara untuk mencapai konfigurasi akhir.

### Subsoal 1 (8 poin)

- Hanya terdapat sebuah kasus uji, dan dapat diunduh [di sini](#).

### Subsoal 2 (10 poin)

- Hanya terdapat sebuah kasus uji, dan dapat diunduh [di sini](#).

### Subsoal 3 (16 poin)

- $P_1 = 2$
- $P_2 = 2$
- $P_3 = 1$
- $P_4 = 1$
- $P_5 = 1$
- $P_6 = 1$

### Subsoal 4 (24 poin)

- $P_1 = 2$
- $P_2 = 1$
- $P_3 = 1$
- $P_4 = 1$
- $P_5 = 1$

- $P_6 = 1$

**Subsoal 5 (42 poin)**

- Untuk setiap  $i$ ,  $1 \leq P_i \leq 6$ .
- Mungkin ada dua atau lebih muka sisi dengan nomor yang sama.