



Hak Cipta
Dilindungi Undang-undang

OLIMPIADE SAINS NASIONAL 2016

DESKRIPSI SOAL



INFORMATIKA/KOMPUTER

Hari 1

- 1. Pasar 16 Ilir**
- 2. Menjinakkan Bom**
- 3. Pos Wisata Sungai**

Waktu: 5 Jam

Hari 1 / Soal 1 - Pasar 16 Ilir

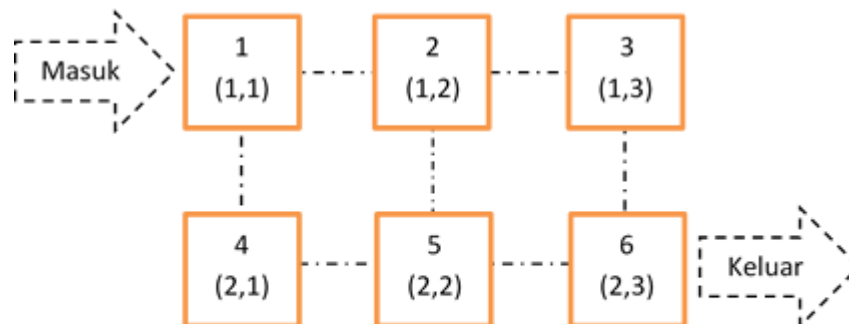
Time limit: 1500 ms

Memory limit: 64 MB

Deskripsi

Dalam perjalanannya menuju Sungai Musi, Pak Dengklek suka melalui Pasar 16 Ilir yang dikenal sebagai pasar yang menjual berbagai jenis panganan lezat. Setiap toko menjual panganan yang berbeda, dan ia sudah memiliki daftar harga lengkap dari semuanya. Beruntungnya, beberapa pemilik toko adalah kerabat Pak Dengklek yang apabila didatangi selain akan memberikan panganannya secara gratis, juga akan memberinya sejumlah tertentu uang.

Toko-toko di Pasar 16 Ilir diatur secara geometris membentuk grid dari $N \times M$ persegi, satu persegi per toko, dan dinomori sebagai (b, k) dengan b adalah nomor baris dan k adalah nomor kolom dalam grid. Jadi, toko di pojok utara-barat dinomori $(1,1)$ dan toko di pojok selatan timur dinomori (N, M) . Gambar 1 mengilustrasikan adanya 6 toko dalam grid 2×3 .



Gambar 1. Ilustrasi toko-toko dalam grid berukuran 2×3

Jika pemilik toko itu adalah kerabatnya: Pak Dengklek akan mendapatkan satu panganan dan menerima sejumlah uang. Jika pemilik toko itu bukan kerabatnya: Pak Dengklek akan membeli satu panganan dan membayar harganya. Dalam perjalanannya di dalam Pasar, Pak Dengklek selalu tiba pertama kali di toko $(1,1)$ dan berakhir di toko (N,M) karena Sungai Musi berada tepat setelah toko (N,M) itu. Dari suatu toko ia akan berjalan ke toko berikutnya dengan mengikuti arah: utara, selatan, timur, atau barat. Di setiap toko yang dilaluinya, satu dari dua kemungkinan kasus akan terjadi:

Bilangan yang tertulis dalam setiap kotak di Gambar 1 menyatakan besarnya uang yang harus dikeluarkan jika Pak Dengklek melalui toko itu. Bilangan positif menyatakan harga panganan

disitu yang harus dibayar Pak Dengklek bilangan negatif menyatakan uang yang diberikan kerabat pemilik toko itu kepada Pak Dengklek selain panganan gratis.

Setiap panganan itu enak sehingga Pak Dengklek ingin selalu mendapatkan panganan-panganan di setiap toko yang dilaluinya. Tetapi, ia telah membatasi pengeluaran untuk mendapatkannya antara P sampai dengan Q rupiah. Selain itu, karena keterbatasan waktunya, ia tidak akan mengunjungi toko yang sama lebih dari sekali.

Tugas Anda adalah membantu Pak Dengklek menghitung berapa banyak cara untuk pergi dari (1, 1) ke (N, M) sambil menghabiskan sejumlah uang, minimal P rupiah dan maksimal Q rupiah.

Total uang yang dihabiskan adalah banyaknya uang yang dikeluarkan untuk membeli makanan dikurangi banyaknya uang yang diterima saat mengunjungi kerabatnya. Karena bisa saja memperoleh uang dalam jumlah yang lumayan banyak, total pengeluaran bisa saja kurang dari 0.

Format Masukan

Baris pertama berisi dua buah bilangan bulat N dan M. N baris berikutnya berisi M buah bilangan bulat yang menyatakan pengeluaran Pak Dengklek di posisi tersebut. Jika $Toko(i,j) \leq 0$, Pak Dengklek mengunjungi tempat kerabatnya sehingga ia tidak perlu membayar makanan bahkan memperoleh uang sumbangan dari mereka.

Baris berikutnya berisi sebuah bilangan bulat K, banyaknya pertanyaan Pak Dengklek. K baris selanjutnya masing-masing berisi dua buah bilangan bulat P dan Q, yaitu kisaran total (minimum dan maksimum, inklusif) uang yang akan dihabiskan Pak Dengklek.

Format Keluaran

Keluaran terdiri dari K baris yang masing-masing berisi tepat sebuah bilangan bulat yang merupakan jawaban dari pertanyaan Pak Dengklek yang ke-i.

Contoh Masukan

0.....6789

2 3

1 2 3

4 5 6

2

0 100

12 12

Contoh Keluaran

4

1

Penjelasan

Gambar 1 mengilustrasikan contoh masukan di atas. Terdapat 4 jalur berbeda yang dapat dilalui untuk pergi dari (1, 1) ke (2, 3) yaitu:

- 1 - 4 - 5 - 6 dengan jumlah 16
- 1 - 2 - 5 - 6 dengan jumlah 14
- 1 - 2 - 3 - 6 dengan jumlah 12
- 1 - 4 - 5 - 2 - 3 - 6 dengan jumlah 21

Saat menjawab pertanyaan pertama, karena keempat jalur tersebut mempunyai jumlah dalam rentang 0 sampai dengan 100, maka keluarkan 4.

Saat menjawab pertanyaan kedua, hanya terdapat satu jalur yang mempunyai jumlah **tepat** 12, maka keluarkan 1.

Subsoal

Untuk semua subsoal berlaku:

- $P \leq Q$
- P dan Q dapat ditampung dalam 64-bit signed integer (long long untuk C/C++ dan int64 untuk pascal).

Subsoal 1 (7 Poin)

Hanya terdiri atas kasus uji berikut ini:

.1.....89

5 2

1 -5

2 3

10 1

-1 2

2 -1

3

0 20

8 15

13 20

Subsoal 2 (8 Poin)

Hanya terdiri atas kasus uji berikut ini:

..2.....89

3 3

10 -5 10

-2 3 1

10 1 8

5

0 20

1 10

0 100

32 35

16 21

Subsoal 3 (6 Poin)

- $N = 1$
- $M \leq 36$
- $-10 \leq \text{Toko}(i,j) \leq 10$
- $K = 1$

Subsoal 4 (10 Poin)

- $1 \leq N * M \leq 10$
- Toko(i,j) berbentuk 2^x dengan $0 \leq x \leq 10$
- Toko(i,j) unik
- $1 \leq K \leq 1.000$

Subsoal 5 (10 Poin)

- $1 \leq N * M \leq 10$
- Toko(i,j) berbentuk 2^x dengan $0 \leq x \leq 10$
- Toko(i,j) unik
- $1 \leq K \leq 100.000$

Subsoal 6 (11 Poin)

- $1 \leq N * M \leq 10$
- $0 \leq \text{Toko}(i,j) \leq 1.000$
- $0 \leq \text{sum}(\text{Toko}(i,j)) \leq 1.000$
- $1 \leq K \leq 1.000$

Subsoal 7 (11 Poin)

- $1 \leq N * M \leq 10$
- $0 \leq \text{Toko}(i,j) \leq 1.000$
- $0 \leq \text{sum}(\text{Toko}(i,j)) \leq 1.000$
- $1 \leq K \leq 100.000$

Subsoal 8 (15 Poin)

- $1 \leq N * M \leq 20$
- $1 \leq K \leq 1.000$
- $-10^9 \leq \text{Toko}(i,j) \leq 10^9$

Subsoal 9 (22 Poin)

- $1 \leq N * M \leq 36$
- $1 \leq K \leq 100.000$
- $-10^9 \leq \text{Toko}(i,j) \leq 10^9$

Peringatan

- Bagi pengguna C++, disarankan menggunakan scanf/printf daripada cin/cout.

Hari 1 / Soal 2 - Menjinakkan Bom

Time limit: 100 ms

Memory limit: 16 MB

Deskripsi

Jembatan Ampera, *landmark* terkenal kota Palembang, dalam bahaya! Seorang teroris telah menanamkan bom tepat di tengah-tengah jembatan Ampera. Bom itu tidak dapat disingkirkan dan bisa meledak kapan saja!

Bom memiliki N buah tombol yang bernomor 1 hingga N . Bom akan meledak tepat setelah penekanan tombol-tombol itu (mana saja) sebanyak T kali, kecuali dijinakkan dengan hal berikut ini.

Di antara tombol-tombol itu terdapat sebuah tombol, sebut saja X . Bom telah dirancang sehingga setelah X ditekan, ia akan berbunyi "BIP" tetapi dengan penundaan sebanyak K kali penekanan tombol berikutnya. Dengan kata lain, jika X ditekan pada penekanan tombol ke- i , suara "BIP"-nya baru akan terdengar pada penekanan tombol ke- $(i+K)$. Harga K adalah antara 0 sampai dengan $N-1$. **Nilai K ini hanya diketahui oleh teroris yang memasang bom.**

Apabila bom telah berbunyi "BIP" sebanyak N kali (tidak harus berturut-turut), bom akan dijinakkan (di-nonaktifkan) asalkan banyaknya penekanan total belum melebihi T kali.

Diketahui juga terdapat dua jenis bom semacam ini: *type-0* dan *type-1*. Bom *type-0* adalah persis seperti yang dijelaskan di atas. Bom *type-1* memiliki tambahan perilaku yaitu akan meledak jika setelah suatu "BIP" terdengar, bunyi "BIP" berikutnya belum juga terdengar dalam N penekanan berikutnya.

Pak Dengklek adalah salah satu penjinak bom yang handal, namun kasus ini tidak semudah yang dikiranya. Tugas anda adalah membantunya menemukan langkah-langkah penekanan yang benar untuk menjinakkan bom ini.

Informasi Tipe Soal

Tipe soal seperti ini biasa disebut "interaktif". Pada soal ini Anda akan berinteraksi dengan program penguji melalui standard input dan standard output. Perhatikan format interaksi di bawah ini dengan saksama.

Format Interaksi

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.

Selanjutnya, program Anda akan menerima input bilangan-bilangan *N*, *T*, dan *R*. Bilangan *N* dan *T* adalah seperti dijelaskan di atas. Bilangan *R* berharga 0 jika bom merupakan bom *type-0*, atau 1 jika bom merupakan bom *type-1*.

Program Anda lalu diminta untuk mengeluarkan sebuah angka antara 1 sampai dengan *N*, yang berarti anda menekan tombol dengan nomor tersebut. Program juri akan menjawab "BIP" apabila setelah penekanan tombol tersebut bom mengeluarkan bunyi "BIP" atau program juri akan menjawab "HENING" apabila tidak. Selama bom belum berhasil dijinakkan atau pun belum meledak, Anda tetap diminta untuk kembali menekan tombol.

Contoh Interaksi

Output Program Juri	Output Program Peserta	Penjelasan
0.....		
4 20 0		
	3	Meskipun tombol X = 3 ditekan ...
HENING		... bom tidak langsung mengeluarkan bunyi BIP, ...
	2	
HENING		

	3	
BIP		... melainkan berbunyi pada penekanan ke-K = 2 berikutnya.
	4	
HENING		
	1	Perhatikan bahwa tidak harus menekan tombol X,...
BIP		... bunyi BIP tetap terdengar akibat penekanan ke-K sebelumnya.
	4	
HENING		
	3	
HENING		
	1	
HENING		
	1	
BIP		
	1	
HENING		
	1	
HENING		
	3	Meskipun sudah $N = 4$ kali menekan tombol X, ...
HENING		... tetap saja bom belum dijinakkan.
	3	
HENING		Hening selama N kali berturut-turut, ...
	3	... BOM AKAN MELEDAK jika bom merupakan bom <i>type -1</i> .

BIP		Bom akhirnya dijinakkan setelah bunyi BIP yang ke-N.
(interaksi selesai)		

Program Anda berhasil apabila (semuanya harus dipenuhi):

- Bom dapat dijinakkan karena telah mengeluarkan bunyi “BIP” sebanyak N kali.
- Banyaknya penekanan tombol yang dilakukan tidak lebih dari T kali.
- Untuk bom yang merupakan bom *type-1*, setelah terdengar bunyi “BIP”, pada maksimal N penekanan tombol kemudian bunyi “BIP” berikutnya harus terdengar lagi.
- Program berhenti sebelum batas waktu.

Program Anda tidak berhasil apabila (cukup salah satu terpenuhi):

- Program Anda tidak mengeluarkan sebuah bilangan bulat atau bilangan bulat yang dikeluarkan tidak berada di antara 1 sampai dengan N.
- Program Anda melakukan penekanan tombol lebih dari T kali.
- Bom belum berhasil dijinakkan karena belum mengeluarkan bunyi “BIP” sebanyak N kali.
- Untuk bom yang merupakan bom *type-1*, ada bunyi “BIP” yang diikuti dengan “HENING” sebanyak N kali. Dengan kata lain, pada N penekanan tombol setelah bunyi “BIP” tersebut, tidak terdengar bunyi “BIP” lagi.
- Program tidak berhenti sebelum batas waktu, termasuk akibat menunggu masukan dari program juri padahal program juri sudah berhenti.

Subsoal

Pada setiap subsoal, berlaku

- $1 \leq X \leq N$
- $0 \leq K \leq N - 1$

Subsoal 1 (8 poin)

- $N = 5$
- $T = 30$

- $R = 0$
- Permainan bisa dimainkan di [sini](#).

Subsoal 2 (9 poin)

- $N = 6$
- $T = 42$
- $R = 1$
- Permainan bisa dimainkan di [sini](#).

Subsoal 3 (8 poin)

- $7 \leq N \leq 36$
- $T = 37 N$
- $0 \leq R \leq 1$

Subsoal 4 (10 poin)

- $7 \leq N \leq 36$
- $T = 21 N$
- $0 \leq R \leq 1$

Subsoal 5 (11 poin)

- $7 \leq N \leq 36$
- $T = 14 N$
- $R = 0$

Subsoal 6 (13 poin)

- $7 \leq N \leq 36$

- $T = 14 \text{ N}$
- $R = 1$

Subsoal 7 (16 poin)

- $7 \leq N \leq 36$
- $T = 5 \text{ N}$
- $R = 0$

Subsoal 8 (25 poin)

- $7 \leq N \leq 36$
- $T = 5 \text{ N}$
- $R = 1$

Hari 1 / Soal 3 - Pos Wisata Sungai

Time limit: 1 s

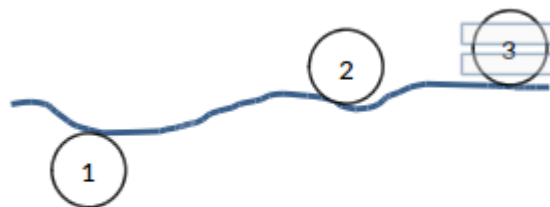
Memory limit: 64 MB

Deskripsi

Pak Dengklek kini menjadi Menteri Pemeliharaan dan Pengembangan Wisata Sungai di Kerajaan Bitania Raya. Di kerajaan ini, semua urusan pengendalian akan dijalankan melalui rangkaian bit-bit (bit adalah binary digit yang bernilai 0 dan 1) serta dengan operasi-operasi bitnya.

Sungai terpanjang di Kerajaan Bitania Raya memiliki M buah pos wisata di sepanjang tepinya, dinomori 1 sampai dengan M . Pos-pos wisata itu dibuka tidak setiap hari. Melainkan, suatu pos wisata yang di suatu hari dibuka, bisa saja di keesokan harinya ditutup atau tetap dibuka, begitu pun sebaliknya. Mekanisme pengaturan pos wisata yang dibuka dan ditutup di suatu hari dilakukan menggunakan rangkaian bit dan operasi-operasinya. Pak Dengklek memiliki prosedur untuk mengaturnya yang sudah terimplementasikan sebagai berikut:

1. Seandainya W merupakan sebuah bilangan biner dengan M digit yang merepresentasikan dibuka atau tidaknya pos-pos wisata pada hari ini. Untuk $1 \leq i \leq M$, digit ke- i dari W akan bernilai 1 jika pos wisata bernomor i dibuka pada hari tersebut, dan akan bernilai 0 jika pos wisata tersebut ditutup.
2. Buat sebuah bilangan biner acak M digit, namakan bilangan biner ini sebagai X .
3. Hitung $W' \leftarrow W \text{ XOR } X$ (catatan: XOR adalah operasi *exclusive-or* bit demi bit antara W dan X . Dalam bahasa C/C++, operasi XOR direpresentasikan sebagai operator \wedge dan dalam bahasa Pascal sebagai xor . Lebih lanjut tentang operasi XOR, lihat catatan di bawah)
4. W' merepresentasikan dibuka atau tidaknya pos-pos wisata pada hari selanjutnya
5. Di hari berikutnya, prosedur ini akan diulangi dengan menggantikan W dengan W' .



Gambar 2. Terdapat 3 pos wisata, pos 1 dan 2 dalam status buka, dan pos 3 tutup.

Gambar 2 menunjukkan adanya 3 pos wisata yang diwakili oleh bilangan biner 3-bit $W=110$. Pos 1 dan pos 2 dalam status buka sehingga bit-bit terkait berharga 1. Sementara, pos 3 dalam status tutup sehingga bit-bit terkait berharga 0.

Pak Dengklek sudah memiliki sederetan N buah bilangan biner $X_1, X_2, X_3, \dots, X_N$ dengan X_i untuk digunakan hari ke- i berikutnya dari sekarang. Ia menjamin, pada hari ke- N dari sekarang, akan ada tepat K buah pos wisata yang dibuka.

Anda penasaran ada berapa kemungkinan deret N buah bilangan biner berbeda yang dimiliki Pak Dengklek jika kondisi pos-pos wisata hari ini diketahui sebagai W sesuai yang dideskripsikan di butir 1 prosedur di atas. Catatan, dua buah deret N bilangan A_1, A_2, \dots, A_N dan B_1, B_2, \dots, B_N dikatakan berbeda jika ada minimal satu harga j untuk $1 \leq j \leq N$, di mana $A_j \neq B_j$. Saat untuk situasi pada Gambar 2, $W = 110$, akan diberikan deret berisi **satu** bilangan biner dan diharapkan tepat ada 1 pos saja yang buka maka banyaknya kemungkinan deret ada 3 yaitu $\{010\}$, $\{100\}$, dan $\{111\}$.

Jawaban Anda, misalnya Z , bisa merupakan bilangan yang sangat besar. Untuk menyederhanakan, tuliskan saja hasil dari $(Z \bmod 1.000.000.007)$ atau $(Z \bmod (10^9 + 7))$ sebagai keluaran Anda. Penjelasan mengenai mod dapat dilihat pada bagian Catatan.

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 akan terdiri dari 3 buah bilangan N , M , dan K .

Baris ketiga akan berisi sebuah bilangan biner W dengan banyaknya bit M .

Format Keluaran

Sebuah baris berisi banyaknya kemungkinan himpunan N buah bilangan biner berbeda yang dimiliki Pak Dengklek. Keluarkan jawaban tersebut dalam **modulo** 1.000.000.007 (atau $10^9 + 7$).

Contoh Masukan 1

0..3456789

1 3 1

110

Contoh Keluaran 1

3

Contoh Masukan 2

0..3456789

2 3 3

110

Contoh Keluaran 2

8

Penjelasan

Untuk contoh masukan 1, ketiga himpunan bilangan tersebut adalah {010}, {100}, dan {111}.

Untuk contoh masukan 2, kedelapan himpunan bilangan tersebut adalah {000, 001}, {001, 000}, {010, 011}, {011, 010}, {100, 101}, {101, 100}, {110, 111}, dan {111,110}.

Subsoal

Pada semua subsoal berlaku:

- $0 \leq K \leq M$

Subsoal 1 (6 poin)

Hanya terdiri dari kasus uji berikut ini:

.1.3456789

2 2 1

10

Subsoal 2 (8 poin)

Hanya terdiri dari kasus uji berikut ini:

..2.456789

3 3 2

101

Subsoal 3 (8 poin)

- $1 \leq N \leq 2$
- $1 \leq M \leq 5$

Subsoal 4 (9 poin)

- $1 \leq N \leq 10$
- $1 \leq M \leq 10$

Subsoal 5 (13 poin)

- $1 \leq N \leq 50$

- $1 \leq M \leq 50$

Subsoal 6 (10 poin)

- $1 \leq N \leq 100$
- $1 \leq M \leq 100$

Subsoal 7 (15 poin)

- $1 \leq N \leq 1.000$
- $1 \leq M \leq 1.000$

Subsoal 8 (14 poin)

- $1 \leq N \leq 100.000$
- $1 \leq M \leq 100.000$
- $0 \leq K \leq 3$

Subsoal 9 (17 poin)

- $1 \leq N \leq 1.000.000.000$
- $1 \leq M \leq 1.000.000$
- $0 \leq K \leq 3$

Catatan

Hasil dari operasi $C \leftarrow A \text{ XOR } B$ adalah digit ke-i dari C akan bernilai 1 jika digit ke-i dari A dan digit ke-i dari B berbeda, dan akan bernilai 0 jika digit ke-i dari A dan digit ke-i dari B sama. Sebagai contoh jika A adalah 111000 dan B adalah 110110, maka nilai C adalah 001110.

Hasil dari operasi $A \text{ modulo } B$ menghasilkan sisa pembagian A oleh B. Sebagai contoh jika A adalah 15 dan B adalah 4, maka $A \text{ mod } B$ adalah 3.