



OLIMPIADE SAINS NASIONAL 2016

DESKRIPSI SOAL



INFORMATIKA/KOMPUTER

Hari 0 (Sesi Latihan)

- 1. Empek-empek**
- 2. Gunting Kertas**
- 3. Matriks Biner**

Waktu: 2 Jam

Hari 0 / Soal 1 - Empek-empek

Batas Memori	: 64 MB
Batas Waktu	: 1 s

Deskripsi

Setelah Pak Dengklek berkonsultasi dengan kerabatnya yang merupakan orang asli Palembang, Pak Dengklek mengetahui bahwa untuk membuat empek-empek diperlukan 3 jenis barang: ikan, tepung sagu, dan gula aren.

Terdapat sebuah pasar dengan N toko yang berada dalam satu barisan dari kiri ke kanan. Toko-toko tersebut ada 3 macam, yaitu yang dinomori dengan 1 menjual ikan, nomor 2 menjual tepung sagu, dan nomor 3 menjual gula aren.

Pak Dengklek ingin berjalan dari kiri ke kanan pasar tersebut. Tetapi, Pak Dengklek tetap ingin memenuhi kebutuhannya sesuai urutan awalnya, yaitu ikan, tepung sagu, lalu gula aren. Dengan kata lain, pada awalnya Pak Dengklek akan mencari sebuah toko yang menjual ikan, setelah itu tepung sagu, dan terakhir gula aren.

Terdapat sangat banyak kemungkinan penentuan cara pemilihan toko sesuai alur yang diinginkan oleh Pak Dengklek, sehingga Anda pun dibuat penasaran: Ada berapa banyak cara menentukan toko yang menjual ikan, kemudian tepung sagu, dan terakhir gula aren?

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 sebuah bilangan bulat N yang menyatakan banyaknya toko.

Baris berikutnya sebuah string berisi N digit yang merepresentasikan toko dari kiri ke kanan. Digit tersebut berkisar dari 1 sampai 3.

Format Keluaran

Keluarkan sebuah bilangan bulat yang menyatakan banyaknya kemungkinan kombinasi urutan toko yang dikunjungi oleh Pak Dengklek.

Contoh Masukan

0..345
7
1121332

Contoh Keluaran

4

Penjelasan

Ada 4 kombinasi urutan toko yang dikunjungi oleh Pak Dengklek, yakni:

- Toko ke-1, toko ke-3, lalu toko ke-5.
- Toko ke-1, toko ke-3, lalu toko ke-6.
- Toko ke-2, toko ke-3, lalu toko ke-5.
- Toko ke-2, toko ke-3, lalu toko ke-6.

Subsoal

Subsoal 1 (7 poin)

Hanya terdiri dari kasus uji berikut:

.1.345

7

1213233

Subsoal 2 (13 poin)

Hanya terdiri dari kasus uji berikut:

..2345

10

1211312323

Subsoal 3 (20 poin)

- $1 \leq N \leq 100$

Subsoal 4 (25 poin)

- $1 \leq N \leq 1.000$

Subsoal 5 (35 poin)

- $1 \leq N \leq 100.000$

Hari 0 / Soal 2 - Gunting Kertas

Batas Memori	: 16 MB
Batas Waktu	: 100 ms

Deskripsi

Pak Dengklek adalah orang yang sangat suka berpikir. Suatu hari, Pak Dengklek terpikir akan sebuah permainan sebagai berikut:

- Permainan tersebut dimainkan oleh 2 orang.
- Kepada pemain pertama, akan diberikan sebuah kertas berukuran $N \times M$ (N dan M bilangan bulat). Pada setiap giliran, pemain yang bermain pada giliran tersebut harus memotong kertas tersebut menjadi dua dan membuang salah satu potongan.
- Potongan kertas yang tidak dibuang harus berukuran $p \times q$ dengan p dan q yang dipilih harus merupakan bilangan bulat positif, di mana salah satu kondisi berikut harus terpenuhi:
 - $p = N$ dan $q < M$, atau
 - $p < N$ dan $q = M$.
- Potongan yang tidak dibuang kemudian diberikan kepada pemain yang lain. Pemain tersebut akan bermain pada giliran berikutnya.

Pemain yang kalah adalah pemain yang tidak dapat memotong kertas yang memenuhi kriteria $p \times q$ di atas, atau pemain yang menerima potongan kertas berukuran 1×1 (ukuran terkecil kertas yang mungkin). Dengan kata lain, pemenang dari permainan ini adalah pemain yang dapat memotong kertas menjadi ukuran 1×1 .

Saat ini, Anda ditantang oleh Pak Dengklek untuk memainkan permainan ini. Dapatkah Anda mengalahkan Pak Dengklek dalam permainan ini?

Format Interaksi

Pada soal ini anda akan berinteraksi dengan juri.

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 akan menerima dua buah bilangan *N* dan *M* (sesuai deskripsi diatas).

Lalu, secara berganti-gantian, program Anda dan program juri akan mengeluarkan dua buah bilangan *p* dan *q*, yang berarti memberikan kertas berukuran *p* x *q* ke lawan. Permainan berakhir apabila program Anda mengeluarkan nilai *p* dan *q* yang tidak valid atau salah satu dari program Anda dan program juri berhasil memotong kertas menjadi ukuran 1 x 1.

Contoh Interaksi 1

Keluaran Program Anda	Keluaran Program Grader
	0..345 5 4
5 3	
	3 3
3 2	
	3 1
1 1	
	(interaksi selesai)

Anda menang dan anda mendapatkan poin untuk kasus uji ini.

Contoh Interaksi 2

Keluaran Program Anda	Keluaran Program Grader
	0..345 5 4
5 3	
	3 3
3 4	
	(interaksi selesai)

Program diberhentikan karena nilai p dan q tidak valid. Anda tidak mendapatkan poin untuk soal ini.

Anda mendapat nilai apabila:

- Anda berhasil memotong kertas menjadi ukuran 1 x 1, dan
- Anda tidak mengeluarkan ukuran yang tidak valid

Anda tidak mendapat nilai apabila:

- Juri berhasil memotong kertas menjadi ukuran 1 x 1, atau
- Anda mengeluarkan ukuran yang tidak valid

Subsoal

- Masukan N dan M diberikan sedemikian rupa sehingga dijamin Anda dapat memenangkan permainan apabila anda bermain secara optimal meskipun Pak Dengklek bermain secara optimal.
- Pada subsoal 1, subsoal 2, serta contoh interaksi, Pak Dengklek tidak selalu bermain optimal. Namun pada subsoal lainnya (subsoal 3 hingga 5), Pak Dengklek akan selalu bermain optimal.

Subsoal 1 (15 poin)

- $N = 3$
- $M = 8$
- Anda dapat memainkan permainan di sini.

Subsoal 2 (15 poin)

- $N = 6$
- $M = 8$
- Anda dapat memainkan permainan di sini.

Subsoal 3 (20 poin)

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

Subsoal 4 (20 poin)

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

Subsoal 5 (30 poin)

- $1 \leq N, M \leq 10.000$

Hari 0 / Soal 3 - Matriks Biner

Batas Memori	: 0 MB
Batas Waktu	: 0 s

Deskripsi

Matriks biner merupakan sebuah matriks yang setiap elemennya merupakan angka 1 atau 0. Karena matriks ini sangat sederhana, maka Pak Dengklek memutuskan untuk menggunakan matriks biner berukuran $N * M$ sebagai password untuk membuka dokumen yang sangat rahasia, "Konspirasi Menaklukkan Dunia".

Namun, karena sudah lama tidak membuka dokumen tersebut, Pak Dengklek lupa password rahasianya! Setelah sehari-hari berusaha mengingat passwordnya kembali, yang dapat diingat oleh Pak Dengklek adalah hasil XOR dari setiap baris dan kolom dari matriks tersebut. Serta, tepat 50% dari elemen matriks tersebut merupakan angka 1 dan sisanya merupakan angka 0. Dengan kata lain, banyak angka 1 dan banyak angka 0 pada matriks tersebut sama.

Hasil XOR dari semua elemen pada baris ke- i adalah R_i .

Hasil XOR dari semua elemen pada kolom ke- j adalah C_j .

Bantulah Pak Dengklek menemukan password rahasianya!

Informasi Tipe Soal

Tipe soal seperti ini biasa disebut "*output-only*". Pada soal ini Anda diminta untuk langsung menuliskan keluaran program ke dalam sebuah berkas keluaran per kasus uji. Setelah itu, kompres semua berkas keluaran dalam sebuah berkas **.zip**.

Masukan untuk soal ini dapat diunduh di [sini](#).

Di dalam berkas zip tersebut ada 1 + 5 masukan untuk diselesaikan: osn-2016-matriks-biner_sample_1.in, osn-2016-matriks-biner_1_1.in, osn-2016-matriks-biner_2_1.in, ..., osn-2016-matriks-biner_5_1.in. Masukan sample tidak dinilai. Untuk setiap berkas masukan yang diselesaikan (Anda tidak harus menyelesaikan semua masukan), buatlah berkas keluaran dengan nama osn-2016-matriks-biner_X_1.out, di mana X adalah nomor masukan (atau osn-2016-matriks-biner_sample_X.out untuk sample) sesuai format keluaran. Setelah itu, kompres semua berkas keluaran dalam sebuah berkas .zip.

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 selanjutnya berisi 2 buah bilangan bulat N dan M .

Baris selanjutnya berisi N buah bilangan bulat, bilangan ke- i pada baris tersebut adalah R_i .

Baris selanjutnya berisi M buah bilangan bulat, bilangan ke- i pada baris tersebut adalah C_i .

Format Keluaran

Keluarkan N baris yang masing-masing berisi M buah bilangan 0 atau 1 yang masing-masing dipisahkan oleh sebuah spasi, yang merupakan matriks rahasia Pak Dengklek. Untuk lebih jelasnya, lihat bagian contoh keluaran.

Contoh Masukan

```
0.....
3 6
0 1 0
0 1 1 1 0 0
```

Contoh Keluaran

```
1 1 0 1 1 0
0 1 1 0 1 0
1 1 0 0 0 0
```

Subsoal

Apabila matrix yang Anda berikan memenuhi kriteria Pak Dengklek (banyak angka 1 dan banyak angka 0 pada matriks tersebut sama), nilai anda akan dihitung menggunakan formula berikut:

$$(X / (N + M))^2 * P$$

X adalah banyaknya baris dan kolom pada matriks Anda yang hasil XOR dari elemen-elemennya sesuai dengan ingatan Pak Dengklek. Perhatikan bahwa Anda tidak harus memenuhi setiap R_i dan C_i . P merupakan nilai maksimal yang mungkin Anda dapatkan pada kasus uji tersebut.

Subsoal 1 (15 poin)

- Nama berkas: osn-2016-matriks-biner_1_1.in
- $N = 2$
- $M = 3$

Subsoal 2 (15 poin)

- Nama berkas: osn-2016-matriks-biner_2_1.in
- $N = 3$
- $M = 4$

Subsoal 3 (20 poin)

- Nama berkas: osn-2016-matriks-biner_3_1.in
- $N = 4$
- $M = 6$

Subsoal 4 (25 poin)

- Nama berkas: osn-2016-matriks-biner_4_1.in
- $N = 6$
- $M = 8$

Subsoal 5 (25 poin)

- Nama berkas: osn-2016-matriks-biner_5_1.in
- $N = 10$
- $M = 10$



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

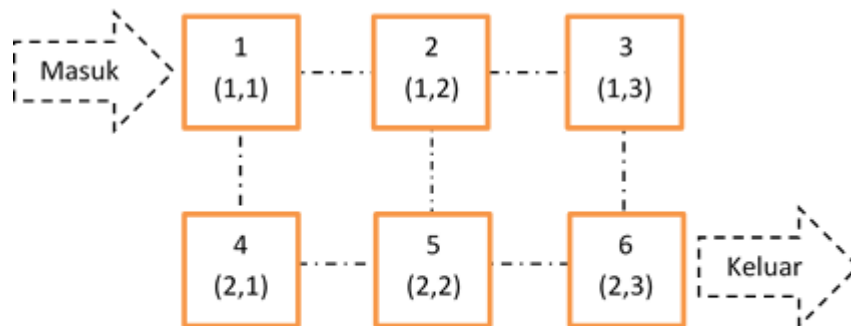
Time limit: 1500 ms

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 penganan gratis.

Setiap penganan itu enak sehingga Pak Dengklek ingin selalu mendapatkan penganan-penganan 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)

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

Subsoal 5 (10 Poin)

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

Subsoal 6 (11 Poin)

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

Subsoal 7 (11 Poin)

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

Subsoal 8 (15 Poin)

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

Subsoal 9 (22 Poin)

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

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

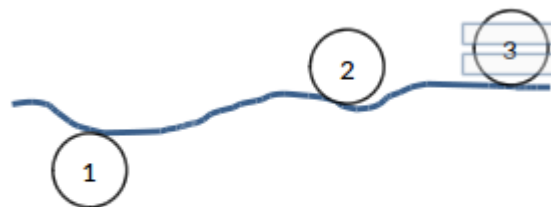
Memory limit: 64 MB

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 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 deretan 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.



OLIMPIADE SAINS NASIONAL 2016

DESKRIPSI SOAL



INFORMATIKA/KOMPUTER

Hari 2

- 1. Robot Pempek**
- 2. Belanja Suvenir**
- 3. Wisata Palembang**

Waktu: 5 Jam

Hari 2/Soal 1 – Robot Pempek

Batas Memori	128 MB
Batas Waktu	0.5 detik

Deskripsi Soal

Pak Dengklek sedang berkunjung ke Kota Palembang. Kota ini terkenal dengan “pempek”, penganan khas terbuat dari ikan yang diambil dari sungai setempat. Ia ingin mencoba membuatnya sendiri, jadi ia berusaha mendapatkan ikan-ikan dari sungai. Teman dekatnya menyarankan Pak Dengklek untuk mendapatkan belida sebagai bahan utama pempek.

Pak Dengklek membawa robot bernama Rempek, ia ingin menguji kemampuan Rempek untuk menangkap ikan-ikan belida. Pak Dengklek memiliki peta berisikan lokasi-lokasi terbaik penangkapan ikan belida. Dalam peta terdapat K posisi terbaik penangkapan ikan belida dimana ikan-ikan biasanya berkumpul. Terdapat K posisi dalam map dan posisi-posisi ini akan digunakan oleh Rempek.

Peta berukuran $N \times M$ dengan posisi paling kiri atas adalah $(1, 1)$ dan kanan bawah adalah (N, M) . Rempek hanya dapat berjalan satu langkah ke empat arah, yaitu dari posisi (r, c) , robot dalam satu langkah dapat berpindah ke salah satu dari $(r - 1, c)$, $(r, c + 1)$, $(r + 1, c)$, $(r, c - 1)$.

Mula-mula Pak Dengklek akan menempatkan Rempek di posisi (x, y) , kemudian Rempek akan mencari lokasi ikan belida terdekat dengan posisinya yang **jaraknya lebih dari 0 langkah**. Satu dari dua kasus ini akan terjadi:

1. Jika terdapat lebih dari satu posisi ikan dengan jumlah langkah minimal dari posisi robot yang sama, maka Rempek berhenti.
2. Jika hanya terdapat satu posisi ikan dengan jumlah langkah minimal dari posisi Rempek:
 - a. Jika lokasi tersebut pernah dikunjungi sebelumnya, Rempek akan berhenti.
 - b. Jika tidak, kunjungi lokasi tersebut, ambil ikan belida sebanyak-banyaknya disitu, kemudian ulangi pencarian lokasi terdekat berikutnya dengan mengulangi prosedur ini.

Ikan di setiap lokasi sangatlah banyak sehingga ketika Rempek mengambil ikan di lokasi manapun, ikan-ikan itu tidak akan pernah habis.

Pak Dengklek ingin mengetahui dimana Rempek posisi terakhir rempek untuk lokasi awal (x,y) . Ia akan menanyakan sejumlah skenario penempatan awal dan untuk setiap skenario anda mendapatkan posisi-posisi akhirnya.

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 4 buah bilangan bulat N, M, K, dan Q.

N baris berikutnya masing-masing berisi tepat M buah karakter, '.' atau 'X'. Karakter 'X' pada baris ke-i kolom ke-j menyatakan adanya lokasi ikan belida di posisi (i, j). Banyaknya karakter 'X' akan sama dengan K.

Q baris berikutnya masing-masing berisi dua buah bilangan bulat x dan y, yang menyatakan skenario penempatan posisi awal Rempek oleh Pak Dengklek. Dijamin bahwa posisi awal ini Rempek **bukan** lokasi keberadaan ikan. Satu baris merupakan posisi awal dari satu skenario. Jadi terdapat Q skenario yang masing-masing harus dijawab. Di awal setiap skenario setiap lokasi ikan adalah belum dikunjungi.

Format Output

Untuk setiap skenario penempatan posisi awal Rempek, keluarkan jawaban yang diminta dalam 1 baris.

Contoh masukan

```
0...4567
10 10 6 3
X.....
.X.....
..X.....
.....
.....
.....X....
.....
.....
.....X....
.....X....
6 5
4 4
9 5
```

Contoh Keluaran

```
10 5
2 2
9 5
```

Penjelasan

Peta yang diberikan adalah sebagai berikut:

	1	2	3	4	5	6	7	8	9	10
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										

Petak berwarna kuning menandakan lokasi ikan belida, sementara petak yang diberi nomor adalah posisi awal Rempek, sesuai pada urutan skenario dalam masukan.

Pada skenario pertama, Rempek berada pada (6, 5). Lokasi ikan terdekat adalah pada (7, 6), kemudian lokasi ikan terdekat dari (7, 6) adalah (9, 6). Lokasi berikutnya yang terdekat adalah (10, 5). Dari posisi tersebut, lokasi ikan terdekatnya adalah (9, 6), tetapi sudah pernah dikunjungi sebelumnya, maka Rempek berhenti di (10, 5).

Pada skenario kedua, Rempek berada pada (4, 4). Selanjutnya Rempek akan bergerak ke (3, 3), kemudian ke (2, 2). Saat berada di (2, 2), terdapat 2 lokasi ikan terdekat (yang sama dekatnya) yaitu (1, 1) dan (3, 3), maka Rempek berhenti di (2, 2).

Pada skenario ketiga, Rempek berada di (9, 5). Dari sini terdapat dua lokasi terdekat dengan banyak langkah minimal yang sama, yaitu (10, 5) dan (9, 6). Maka Rempek berhenti di (9, 5).

Subsoal

Pada semua subsoal, berlaku:

- $1 \leq x \leq N$
- $1 \leq y \leq M$

Subsoal 1

Hanya terdiri atas kasus uji berikut ini:

```

.1..4567
8 10 6 3
.X.....X
.....
.....
.....
.....
.....X....
.....
X...X...X
4 4
1 6
1 5

```

Subsoal 2

Hanya terdiri atas kasus uji berikut ini:

```

..2.4567
20 20 15 10
XX.....
.....X.
.X..X.....
.....
.....X....
.....
...X.....
.....
.....X.
.....
.....X.
.....
.....X....
.....
.....X....
.....
.....X
.....X
.X.....X.X
16 17
1 15
7 9
6 13
18 6

```

```
1 20
4 3
3 1
20 16
17 8
```

Subsoal 3

- $1 \leq N, M, K \leq 200$
- $Q = 1$

Subsoal 4

- $1 \leq N, M, K \leq 200$
- $1 \leq Q \leq 200$

Subsoal 5

- $1 \leq N, M, K \leq 2.000$
- $1 \leq Q \leq 2.000$

Subsoal 6

- $1 \leq N, M, K \leq 2.000$
- $1 \leq Q \leq 300.000$

Subsoal 7

- $1 \leq N, M \leq 2.000$
- $1 \leq K, Q \leq 300.000$

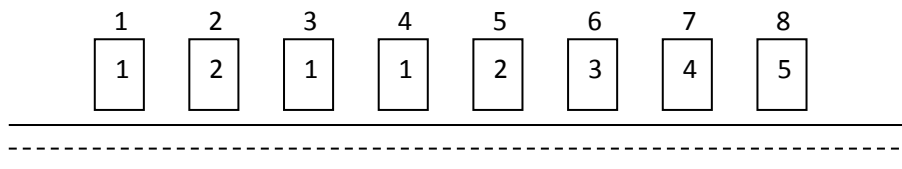
Hari 2/Soal 2: Belanja Suvenir

Batas Memori	128 MB
Batas Waktu	1 detik

Deskripsi Soal

Kwik dan Kwak sedang berlibur di Palembang dengan mengikuti satu rombongan wisata. Hari ini, acara tur akan memberi mereka kesempatan untuk mengunjungi N toko souvenir yang tersebar di sepanjang jalan raya. Jarak antara toko cukup jauh sehingga rombongan akan dibawa mengunjungi satu demi satu toko. Masing-masing toko menjual satu produk khas – toko ke- i menjual satu produk unggulan T_i . Hanya saja, Kwik dan Kwak harus mengerjakan PR mereka bersamaan dengan acara ini, sehingga mereka menerapkan strategi berikut ini agar dapat membeli souvenir (mereka=Kwik dan Kwak):

- Ketika rombongan mengunjungi sebuah toko, maksimum satu saja di antara mereka yang boleh masuk.
- Supaya adil, mereka masing-masing akan masuk ke tepat M toko.
- Masuk ke suatu toko berarti juga membeli produk unggulan di toko itu.
- Mereka masing-masing akan memasuki toko-toko dalam satu rentang (deretan), tanpa satu toko pun yang dilewati.
- Karena mereka akan membeli produk unggulan saja, mereka tidak mau masuk ke toko yang memiliki produk unggulan yang sama dengan yang telah dibeli sebelumnya. Perhatikan bahwa jika Kwik sudah pernah masuk ke toko dengan produk unggulan x , Kwak tetap dapat masuk ke toko dengan produk unggulan x , seandainya Kwak belum pernah membeli produk unggulan x di toko lain.



Gambar 2. Ilustrasi ada 8 toko dan masing-masing produknya (dituliskan di dalam kotak)

Kwik dan Kwak ingin dapat membeli sebanyak mungkin souvenir. Oleh karena itu, mereka ingin memaksimalkan nilai M . Mengingat tur hari ini akan dimulai dalam beberapa menit, bantulah Kwik dan Kwak menentukan toko mana saja yang akan mereka masuki!

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:

- o jika kasus uji tersebut memenuhi batasan subsoal ke-i, maka karakter ke-i berisi i, atau
- o 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 sebuah bilangan bulat N, banyaknya toko. Baris ketiga akan berisi N buah bilangan T_i , produk unggulan toko ke-i.

Format Keluaran

Keluarkan 4 buah bilangan bulat a, b, c, dan d, yang menyatakan rentang-rentang toko-toko yang Kwik dan Kwak akan masuki. Toko yang Kwik masuki adalah toko pada rentang [a,b], sedangkan Kwak akan memasuki toko pada rentang [c,d]. Jika ada lebih dari 1 keluaran yang mungkin, keluarkan salah satu yang mana saja.

Contoh Masukan 1

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

Contoh Keluaran 1

```
1 3 4 6
```

Contoh Masukan 2

```
0..34567
8
1 2 1 1 2 3 4 5
```

Contoh Keluaran 2

```
1 2 7 8
```

Contoh Masukan 3

```
0..34567
10
2 1 1 1 1 2 3 1 2 3
```

Contoh Keluaran 3

```
5 7 8 10
```

Penjelasan

Untuk contoh masukan 1, Kwik masuk ke toko pertama sampai ketiga, dimana ia masuk ke toko dengan produk unggulan [1,2,3]. Sementara itu, Kwak masuk ke toko keempat sampai keenam, dimana ia masuk ke toko dengan produk unggulan [4,5,6].

Untuk contoh masukan 2 (Gambar 2 mengilustrasikannya), Kwik masuk ke toko pertama sampai kedua, dimana ia masuk ke toko dengan produk unggulan [1,2]. Sementara itu, Kwak masuk ke toko ketujuh sampai kedelapan, dimana ia masuk ke toko dengan produk unggulan [4,5]. Kwik tidak dapat masuk ke toko ketiga, karena ia akan memasuki toko dengan produk unggulan yang sudah pernah ia lihat.

Untuk contoh masukan 3, Kwik masuk ke toko kelima sampai ketujuh, dimana ia masuk ke toko dengan produk unggulan [1,2,3]. Sementara itu, Kwak masuk ke toko kedelapan sampai kesepuluh, dimana ia masuk ke toko dengan produk unggulan [1,2,3]. Perhatikan bahwa Kwik dan Kwak boleh membeli produk unggulan yang sama.

Subsoal

Pada semua subsoal, berlaku:

- $2 \leq N \leq 2.000.000$
- $1 \leq T_i \leq N$

Subsoal 1

Hanya terdiri atas kasus uji berikut ini:

```
.1..4567
18
1 6 3 8 10 8 6 4 9 9 1 10 7 7 6 7 2 8
```

Subsoal 2

Hanya terdiri atas kasus uji berikut ini:

..2.4567

45

23 8 22 22 19 2 23 13 7 17 10 18 16 19 2 14 7 4 15 13 16 14 10 23
22 12 17 12 25 13 23 9 21 7 20 12 24 15 13 3 20 11 21 8 2

Subsoal 3

- $2 \leq N \leq 10$

Subsoal 4

- $2 \leq N \leq 50$

Subsoal 5

- $2 \leq N \leq 200$

Subsoal 6

- $2 \leq N \leq 2.000$

Subsoal 7

- $2 \leq N \leq 2.000.000$

Hari 2/Soal 3: Wisata Palembang

Batas Memori	-
Batas Waktu	-

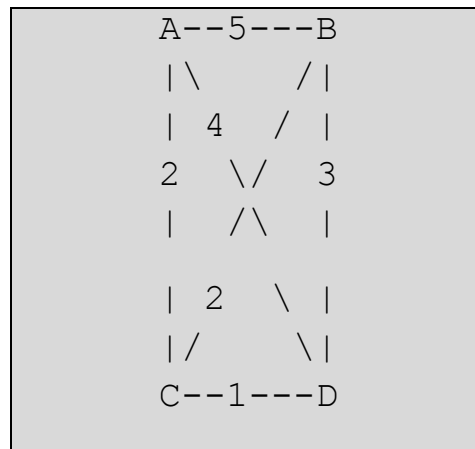
Deskripsi Soal

Pak Dengklek dan bebek-bebeknya senang berwisata kuliner. Mereka bersyukur berkesempatan berada di Kota Palembang yang kaya berbagai lokasi wisata kuliner untuk dikunjungi. Mereka mempersiapkan rencana berkeliling kota berdasarkan suatu daftar berisi $N-1$ lokasi kuliner terfavorit di kota ini. yang dinomori mulai dari 2 sampai dengan N . Nomor 1 adalah hotel mereka sebagai titik mulai dan akhir dari perjalanan mereka. Tabel berikut ini mengilustrasikan suatu contoh list tersebut.

Lokasi	Makanan/Minuman
2	Pempek
3	Tekwan
4	Model
5	Laksan
6	Celimpungan
7	Kerupuk Kemplang
8	Martabak Kari
9	Lapis Legit Maksuba
10	Es Kacang Merah

Sayangnya, informasi jarak ruas jalan antar lokasi tidak tersedia, kecuali bahwa antar setiap lokasi dalam list, termasuk hotel, selalu terdapat ruas jalan dua arah yang menghubungkannya dengan **panjang yang tidak lebih dari L** . Kemudian, dengan informasi itu Pak Dengklek menyusun rute perjalanannya mulai dari lokasi 1 (hotel mereka), lalu mengunjungi $(N-1)$ lokasi kuliner dalam list itu, dalam urutan tertentu, dan berakhir kembali ke hotel. Setiap lokasi kuliner hanya akan dikunjungi tepat sekali.

Jika panjang ruas-ruas jalan itu diketahui tentunya total jarak yang akan ditempuh dapat diketahui, seperti diilustrasikan pada Gambar 3 berikut.



Gambar 3. Contoh Ruas-ruas jalan untuk $N=4$, $M=2$ dan $L=5$.

Maka, dari contoh itu semua kemungkinan rute beserta total-total jaraknya adalah sebagai berikut.

- A--B--C--D--A: $5 + 2 + 1 + 4 = 12$
- A--B--D--C--A: $5 + 3 + 1 + 2 = 11$
- A--C--B--D--A: $2 + 2 + 3 + 4 = 11$
- A--C--D--B--A: $2 + 1 + 3 + 5 = 11$
- A--D--B--C--A: $4 + 3 + 2 + 2 = 11$
- A--D--C--B--A: $4 + 1 + 2 + 5 = 12$

Dari semua rute tersebut, terdapat 2 buah kemungkinan total jarak berbeda, yaitu 11 dan 12. Sekarang, ia sedang terpancing memikirkan pertanyaan kebalikannya. Jika diketahui **banyaknya kemungkinan total jarak berbeda tidak kurang dari M**, dengan N dan L sesuai penjelasan sebelumnya, bisakah ia mendapatkan satu kemungkinan **panjang-panjang ruas jalan yang memenuhi**? Asumsikan bahwa panjang-panjang ruas jalan itu bilangan bulat.

Pak Dengklek meminta pertolongan anda untuk menemukan jawaban yang memenuhi batasan-batasan tersebut (N , M dan L). Ingat bahwa anda hanya menemukan satu kemungkinan saja dan jika jawaban anda memang memenuhi batasan-batasan tersebut anda akan memperoleh nilai penuh. Jika tidak, Pak Dengklek masih memberikan suatu kelonggaran mengenai batasan L untuk mendapatkan nilai parsial selama batasan nilai M terpenuhi. Perhitungan nilai parsial ini ditentukan menurut rumus perhitungan yang dijelaskan di bagian Subsoal (bagian terakhir).

Informasi Tipe Soal

Soal ini adalah soal “output-only”. Dalam soal ini, anda harus menuliskan keluaran program dalam file keluaran untuk setiap kasus uji. Kemudian, compress seluruh file menjadi sebuah file .zip.

Masukan untuk soal ini dapat diunduh **di sini**. Di dalam berkas zip tersebut ada 1 + 8 masukan untuk diselesaikan: **wisata_sample_1.in**, **wisata_1.in**, **wisata_2.in**, ..., **wisata_8.in**. Masukan sample tidak dinilai. Untuk setiap berkas masukan yang diselesaikan (Anda tidak harus menyelesaikan semua masukan), buatlah berkas keluaran dengan nama **wisata_X.out**, di mana X adalah nomor masukan (atau **wisata_sample_X.out** untuk sample) sesuai format keluaran. Setelah itu, kompres semua berkas keluaran dalam sebuah berkas .zip.

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.

Kemudian, baris berikutnya berisi tiga buah bilangan bulat N , M , dan L , yang berturut-turut menyatakan banyaknya lokasi, banyaknya total jarak berbeda yang diinginkan Pak Dengklek, dan parameter penilaian. Nilai L akan digunakan untuk menentukan nilai pada suatu kasus uji. Lihat bagian Subsoal untuk lebih jelasnya.

Format Keluaran

Keluaran terdiri dari N baris, yang menjelaskan panjang jalan yang diinginkan. Baris ke- i berisi N buah bilangan bulat, dengan bilangan ke- j pada baris ini adalah nilai A_{ij} . Nilai A_{ii} harus 0 dan nilai A_{ij} harus sama dengan nilai A_{ji} . Bilangan-bilangan dalam satu baris dipisahkan oleh spasi.

Pasti terdapat setidaknya satu jawaban yang memenuhi persyaratan, yaitu terdapat paling tidak M buah total jarak yang berbeda. Anda dapat mengeluarkan jawaban manapun yang sah, dan nilai Anda akan bergantung kepada keluaran yang Anda hasilkan.

Contoh Masukan

Nama file: wisata_sample_1.in

```
0.....
4 2 5
```

Contoh Keluaran

Nama file: wisata_sample_1.out

```
0 5 2 4
5 0 2 3
2 2 0 1
4 3 1 0
```

Subsoal

Setiap subsoal hanya terdiri atas satu kasus uji. Nilai untuk setiap kasus uji adalah 0 jika ada kriteria berikut yang tidak terpenuhi:

- keluaran sesuai format yang telah dijelaskan pada format keluaran,
- $1 \leq A_{ij} = A_{ji} \leq 10^{17}$ untuk $i, j = 1..N$ dan $i \neq j$,
- $A_{ii} = 0$ untuk $i = 1..N$, dan
- terdapat paling tidak M buah kemungkinan total jarak berbeda di antara semua urutan pengunjungan.

Jika seluruh kriteria di atas telah terpenuhi, maka nilai Anda untuk kasus uji tersebut adalah:

- Kasus uji 1, 2, 3, 4, 5, dan contoh:

$$\min\left(\frac{L}{\max(A_{ij})}, 1\right) \times P$$

- Kasus uji 6, 7, 8:

$$P^{\min\left(\frac{L}{\max(A_{ij})}, 1\right)}$$

dengan P adalah poin maksimum untuk kasus uji yang bersangkutan (P bernilai 100 untuk kasus uji contoh).

Subsoal 1

- **Nama file:** wisata_1.in
- $N = 4$
- $M = 3$
- $L = 2$

Subsoal 2

- **Nama file:** wisata_2.in
- $N = 5$
- $M = 5$
- $L = 2$

Subsoal 3

- **Nama file:** wisata_3.in
- $N = 5$
- $M = 8$
- $L = 3$

Subsoal 4

- **Nama file:** wisata_4.in

- $N = 5$
- $M = 12$
- $L = 6$

Subsoal 5

- **Nama file:** wisata_5.in
- $N = 10$
- $M = 70$
- $L = 9$

Subsoal 6

- **Nama file:** wisata_6.in
- $N = 8$
- $M = 2.520$
- $L = 40.000$

Subsoal 7

- **Nama file:** wisata_7.in
- $N = 9$
- $M = 20.160$
- $L = 2.250.000$

Subtask 8

- **Nama file:** wisata_8.in
- $N = 10$
- $M = 181.440$
- $L = 200.000.000$