

Problem A. Arena of Greed

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **256 megabytes**

Akhir-akhir ini, Pak Chanek suka memainkan sebuah permainan bernama **Arena of Greed**. Sesuai namanya, permainan ini bertujuan untuk mencari orang serakah sejati yang akan dinobatkan sebagai raja di Compfestnesia.

Permainan ini dimainkan oleh dua orang yang saling bergiliran dimana Pak Chanek mendapatkan giliran pertama. Mula-mula, terdapat suatu kotak harta yang berisi N keping emas. Permainan selesai apabila tidak ada lagi emas di dalam kotak. Pada setiap gilirannya, pemain dapat melakukan salah satu dari dua langkah berikut:

- Mengambil satu keping emas dari dalam kotak.
- Mengambil setengah dari sisa emas di dalam kotak dengan syarat sisa emas sebelum diambil haruslah genap.

Tentu saja setiap pemain akan berusaha untuk memaksimalkan total keping emas yang didapatkan. Untuk itu, Pak Chanek meminta bantuan Anda untuk mencari jumlah keping emas yang akan didapat Pak Chanek di akhir permainan apabila Pak Chanek dan lawannya bermain optimal.

Input

Baris pertama berisi sebuah bilangan bulat T ($1 \leq T \leq 10^5$) yang menyatakan banyak kasus uji. T baris selanjutnya masing-masing berisi sebuah bilangan bulat N ($1 \leq N \leq 10^{18}$).

Output

T buah baris masing-masing berisi jawaban yang diminta Pak Chanek.

Example

standard input	standard output
2	2
5	4
6	

Note

Pada kasus pertama, permainan dapat berlangsung seperti di bawah ini:

1. Pak Chanek mengambil 1 keping emas.
2. Lawan Pak Chanek mengambil 2 keping emas.
3. Pak Chanek mengambil 1 keping emas.
4. Lawan Pak Chanek mengambil 1 keping emas.

Pada kasus kedua, permainan dapat berlangsung seperti di bawah ini:

1. Pak Chanek mengambil 3 keping emas.
2. Lawan Pak Chanek mengambil 1 keping emas.

3. Pak Chanek mengambil 1 keping emas.
4. Lawan Pak Chanek mengambil 1 keping emas.

Problem B. Biru Merah Makaraku

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Fakultas kami sedang merayakan dies natalis ke-34! Dalam merayakannya, Fakultas Ilmu Komputer, Universitas Indonesia (Fasilkom), mengadakan lomba CPC - Coloring Pavements Competition. Inti dari lomba CPC adalah mewarnai jalan di Fasilkom dengan warna biru atau merah. Terdapat N tempat menarik dan M jalan dua arah. Jalan ke- i menghubungkan tempat menarik nomor U_i dan nomor V_i , untuk ($1 \leq i \leq M$). Setiap pasang tempat menarik dapat dicapai dengan menggunakan beberapa jalan.

Aturan dari lomba CPC adalah sebagai berikut:

- Perlombaan diikuti oleh dua orang. Satu orang memilih warna biru, dan satu orang lainnya memilih warna merah. Untuk memudahkan, sebut saja kedua orang tersebut *Blue* dan *Red*.
- *Blue* mewarnai jalan dengan warna biru, sedangkan *Red* mewarnai jalan dengan warna merah. Kedua pemain mulai dari tempat menarik nomor 1. Awalnya, semua jalan tidak memiliki warna.
- Pada setiap babak, dari tempat menarik setiap pemain, *Blue* dan *Red* memilih jalan tak berwarna **berbeda** dan berpindah ke tempat menarik selanjutnya melalui jalan yang dipilih secara bersamaan.
- Perlombaan berhenti ketika *Blue* dan *Red* tidak dapat berpindah. Dengan kata lain, tidak ada dua jalan tak berwarna berbeda yang dapat mereka pilih untuk berpindah.

Chaneka tertarik mengikuti lomba tersebut, tetapi ia tidak ingin energinya banyak terbuang. Maka dari itu, Chaneka hanya ingin menghitung banyaknya kemungkinan kondisi warna jalan di Fasilkom setelah lomba CPC selesai. Karena menghitung juga bisa melelahkan, Chaneka meminta bantuan Anda untuk menjawab rasa penasarannya.

Dua kondisi jalan akhir dikatakan berbeda jika terdapat jalan U dengan warna berbeda pada kedua kondisi tersebut.

Input

Baris pertama berisi dua buah bilangan bulat N dan M . N ($2 \leq N \leq 2 \cdot 10^3$) menyatakan banyaknya tempat menarik, M ($1 \leq M \leq 2 \cdot N$) menyatakan banyaknya jalan di Fasilkom. Dijamin untuk setiap tempat menarik kecuali tempat menarik nomor 1 terhubung dengan tepat dua jalan.

M baris berikutnya masing-masing berisi dua buah bilangan bulat U_i dan V_i ($1 \leq U_i, V_i \leq N, U_i \neq V_i$), yang menyatakan tempat menarik yang dihubungkan oleh jalan ke- i .

Dijamin untuk setiap pasang tempat menarik, terdapat rute yang menghubungkannya secara langsung maupun tidak langsung.

Output

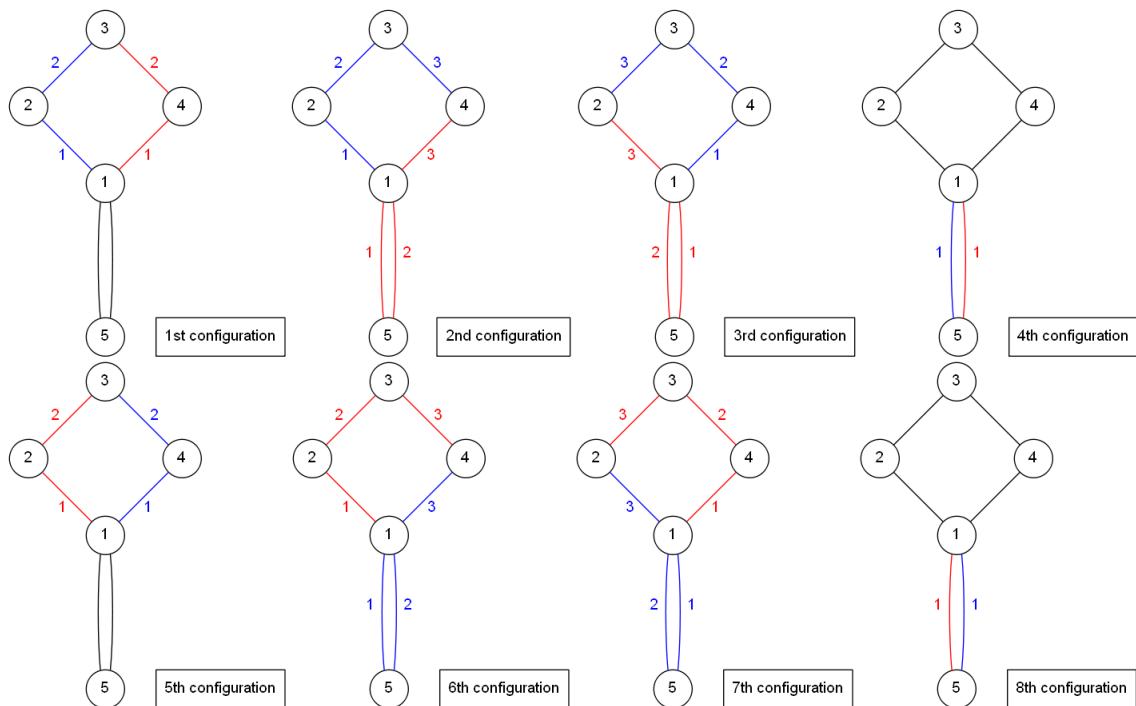
Sebuah baris yang berisi banyaknya kemungkinan kondisi warna jalan di Fasilkom setelah lomba CPC selesai modulo $10^9 + 7$.

Example

standard input	standard output
5 6 1 2 2 3 3 4 4 1 1 5 5 1	8

Note

Seluruh kemungkinan kondisi warna jalan di Fasilkom sesuai contoh masukan adalah sebagai berikut:



Jalan dengan angka berwarna biru merupakan urutan jalan yang dilalui *Blue*, dan jalan dengan angka berwarna merah merupakan urutan jalan yang dilalui *Red*

Problem C. Captain Kuda

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Pak Chanek baru saja memenangkan kompetisi catur nasional dan mendapatkan hadiah papan catur besar berukuran $N \times M$. Pak Chanek yang sudah bosan bermain catur secara konvensional kini mendefinisikan fungsi $F(X, Y)$ yang didefinisikan sebagai jumlah langkah minimal untuk menggerakkan kuda dari petak $(1, 1)$ ke petak (X, Y) . Ternyata menghitung $F(X, Y)$ terlalu mudah, sehingga Pak Chanek mendefinisikan:

$$G(X, Y) = \sum_{i=X}^N \sum_{j=Y}^M F(i, j)$$

Diberikan X dan Y , anda diminta untuk menghitung nilai $G(X, Y)$.

Sebuah kuda dapat bergerak dari petak (a, b) ke petak (a', b') jika dan hanya jika $|a - a'| > 0$, $|b - b'| > 0$, dan $|a - a'| + |b - b'| = 3$. Tentunya kuda tidak boleh keluar dari papan catur.

Input

Baris pertama berisi sebuah bilangan bulat T ($1 \leq T \leq 100$), banyaknya kasus uji.

Setiap kasus uji berisi empat buah bilangan $X Y N M$ ($3 \leq X \leq N \leq 10^9, 3 \leq Y \leq M \leq 10^9$).

Output

Untuk setiap kasus uji, keluarkan sebuah baris berisi nilai $G(X, Y)$

Example

standard input	standard output
2	27
3 4 5 6	70
5 5 8 8	

Problem D. Daerah Ular Gila

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Pak Chanek sang ninja suatu hari ditugaskan misi untuk menangani kasus ular gila yang menyerah suatu daerah. Kini Pak Chanek sudah tiba di bukit dan daerah tujuannya berada tepat di bawah bukit tersebut. Daerah tujuan Pak Chanek dapat direpresentasikan sebagai grid berukuran 1000×1000 . Pada daerah tersebut terdapat N ular gila dimana ular gila ke- i berada di petak (X_i, Y_i) dan memiliki level bahaya B_i . Pak Chanek akan menggunakan rasengan dan jurus seribu bayangan yang telah ia pelajari dari Pak Hokage untuk menyelesaikan misi ini. Strategi menyerangnya sebagai berikut:

1. Pak Chanek membuat M bayangan
2. Setiap bayangan memilih seekor ular gila sebagai target penyerangan. Setiap bayangan wajib memilih ular gila yang berbeda.
3. Semua bayangan Pak Chanek lompat dari bukit dan menyerang target mereka secara bersamaan menggunakan rasengan radius R . Jika ular gila pada petak (X, Y) diserang rasengan secara langsung, dia dan semua ular gila pada petak (X', Y') dimana $\max(|X' - X|, |Y' - Y|) \leq R$ akan mati.
4. Pak Chanek asli menghitung skor penyerangan ini. skor penyerangan didefinisikan sebagai kuadrat dari total bahaya ular-ular yang terbunuh.

Kini Pak Chanek penasaran, berapakah total dari skor untuk setiap kemungkinan penyerangan? karena jawaban mungkin sangat besar, Pak Chanek hanya ingin jawabannya yang di modulo $10^9 + 7$.

Input

Baris pertama tiga bilangan $N M R$ ($1 \leq M \leq N \leq 2 \cdot 10^3, 0 \leq R < 10^3$), banyaknya ular gila pada daerah tersebut, banyaknya bayangan Pak Chanek, dan radius rasengan yang dikeluarkan.

N baris berikutnya berisi tiga bilangan, X_i, Y_i , dan B_i ($1 \leq X_i, Y_i \leq 10^3, 1 \leq B_i \leq 10^6$). Dijamin tidak ada dua ular gila yang menempati petak yang sama.

Output

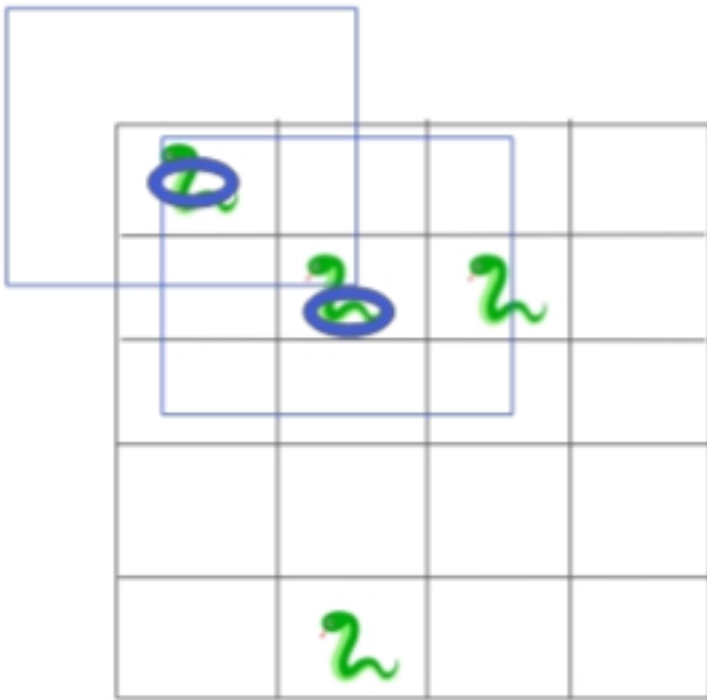
Sebuah baris dengan bilangan bulat positif yang menyatakan total skor untuk setiap kemungkinan penyerangan.

Example

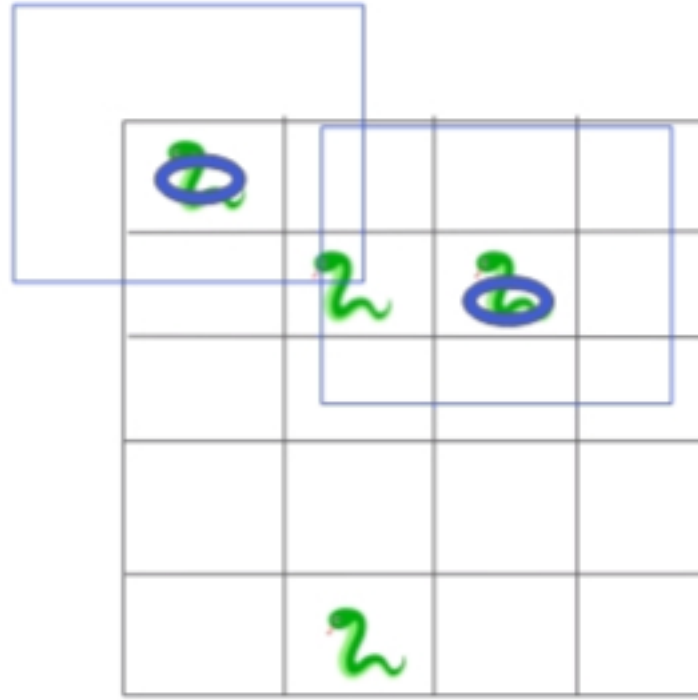
standard input	standard output
4 2 1 1 1 10 2 2 20 2 3 30 4 2 40	33800

Note

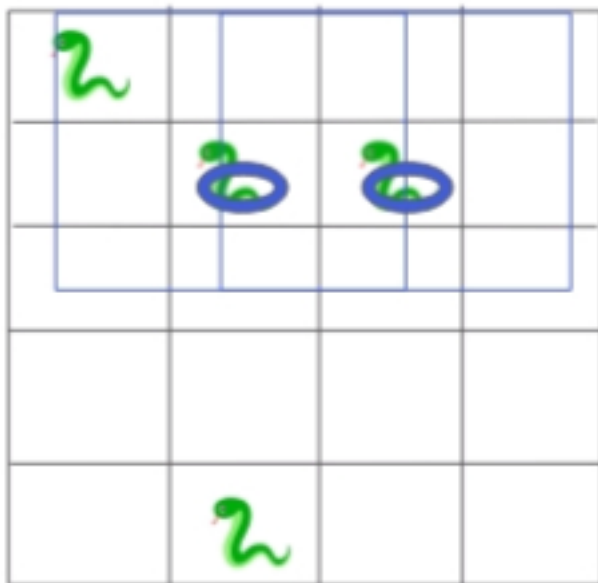
Berikut ilustrasi untuk keenam kemungkinan penyerangan, lingkaran menyatakan pusat serangan dan persegi biru menyatakan daerah serangan rasengan:



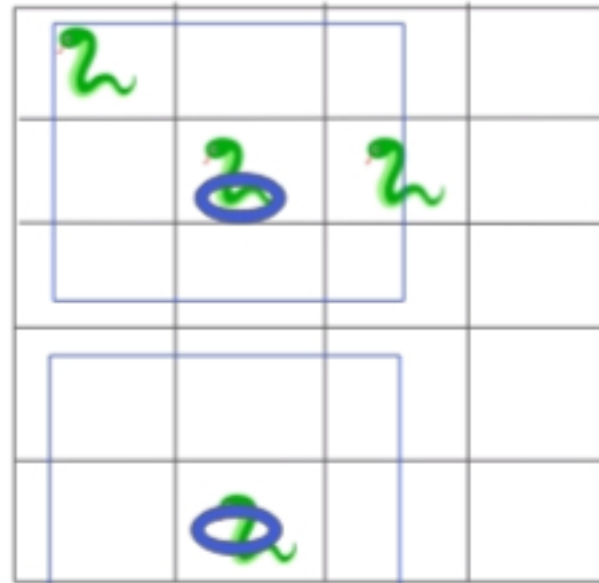
Score: 3600



Score: 3600



Score: 3600



Score: 10000

Sehingga total skor untuk setiap penyerangannya: $3.600 + 3.600 + 4.900 + 3.600 + 10.000 + 8.100 = 33.800$.

Problem E. Eksitasi Atom

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Pak Chanek sedang mengikuti festival teka-teki ilmiah yang sedang marak di perkotaan. Ia menemukan permasalahan menarik di festival tersebut dan ingin menyelesaikannya.

Terdapat N buah atom yang dinomori dari 1 hingga N . Atom yang dimiliki Pak Chanek sangat unik. Pada awalnya, setiap atom berada dalam kondisi normal. Setiap atom dapat di eksitasi. Mengeksitasi atom ke- i membutuhkan D_i energi. Jika atom ke- i tereksitasi, ia akan menghasilkan A_i energi.

Atom-atom ini juga membentuk ikatan satu arah yang unik. Untuk setiap $(1 \leq i < N)$, apabila atom ke- i tereksitasi, maka atom ke- E_i juga akan ikut tereksitasi tanpa membutuhkan energi tambahan. Pada awalnya, $E_i = i + 1$. Perhatikan bahwa atom ke- N tidak dapat membentuk ikatan ke mana pun. Jumlah atom yang dapat kamu eksitasi tidak dibatasi (termasuk 0).

Pak Chanek harus merubah sebanyak **tepat** K buah ikatan atom. Sebanyak tepat K kali, Pak Chanek memilih sebuah atom i , $(1 \leq i < N)$ dan merubah E_i menjadi nilai lain, selain i dan E_i pada saat itu. Perhatikan bahwa ikatan suatu atom dapat tidak dirubah sama sekali, atau pun dirubah lebih dari sekali. Bantulah Pak Chanek untuk menentukan keuntungan energi maksimum yang bisa didapatkannya!

catatan: Kamu harus mengubah ikatan **tepat** K atom terlebih dahulu sebelum dapat mulai mengeksitasi atom.

Input

Baris pertama berisi dua buah bilangan bulat, N dan K ($4 \leq N \leq 10^5, 0 \leq K < N$) yang berturut-turut menyatakan jumlah atom pada awalnya dan jumlah ikatan atom berbeda yang harus dirubah Pak Chanek.

Baris kedua berisi N buah bilangan bulat A_i ($1 \leq A_i \leq 10^6$), yang menyatakan energi yang akan didapatkan apabila atom ke- i berada dalam kondisi tereksitasi.

Baris ketiga berisi N buah bilangan bulat D_i ($1 \leq D_i \leq 10^6$), yang menyatakan biaya energi yang dibutuhkan untuk membuat atom ke- i berada dalam kondisi tereksitasi.

Output

Sebuah baris berisi keuntungan energi maksimum yang bisa didapatkan Pak Chanek.

Example

standard input	standard output
6 1 5 6 7 8 10 2 3 5 6 7 1 10	35

Note

Solusi optimal adalah mengubah E_5 menjadi 1 lalu mengeksitasi atom 5. Ini akan menyebabkan atom 1, 2, 3, 4, 5 menjadi tereksitasi. Total keuntungan energi yang didapatkan: $(5 + 6 + 7 + 8 + 10) - 1 = 35$.

Salah satu kemungkinan lain adalah mengubah E_3 menjadi 1 lalu mengeksitasi atom 3 (yang membuat atom 1, 2, 3 tereksitasi) dan mengeksitasi atom 4 (yang membuat atom 4, 5, 6) tereksitasi. Total keuntungan energi yang didapatkan: $(5 + 6 + 7 + 8 + 10 + 2) - (6 + 7) = 25$. Tidak optimal.

Problem F. Flamingo Misterius

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **256 megabytes**

Ini adalah soal interaktif. Anda harus flush sehabis mencetak setiap baris. Contoh, di C++ memakai `fflush(stdout)`, di Java — `System.out.flush()`, di Pascal — `flush(output)` di Python — `sys.stdout.flush()`.

Pak Chanek ingin membeli seekor flamingo untuk menemani ayam-ayam di peternakannya. Sebelum pergi ke toko binatang, Pak Chanek singgah ke festival binatang untuk bersenang-senang. Ternyata disana, terdapat hiburan permainan berhadiah seekor flamingo.

Terdapat N buah kandang misterius masing-masing dinomori dari 1 hingga N . Kandang ke- i memiliki A_i ($0 \leq A_i \leq 10^3$) ekor flamingo di dalamnya, untuk setiap ($1 \leq i \leq N$). Namun, tentu saja tuan yang mengadakan hiburan tersebut akan merahasiakan jumlah flamingo yang terdapat di dalam setiap kandang. Untuk memenangkan seekor flamingo, Pak Chanek harus menebak ada berapa jumlah flamingo di setiap kandang.

Kebetulan sekali Pak Chanek hanya sempat membeli N buah koin. Nasing-masing koin tersebut dapat digunakan untuk bertanya sekali, ada berapakah banyak flamingo di dalam kandang ke L hingga ke R secara inklusif, dengan $L < R$.

Input

Pada awalnya, juri akan mengeluarkan sebuah baris berisi bilangan bulat N ($3 \leq N \leq 10^3$), yang menyatakan jumlah kandang misterius sekaligus banyak pertanyaan maksimum yang dapat ditanyakan Pak Chanek.

Untuk setiap pertanyaan, juri akan memberikan sebuah bilangan bulat yang menyatakan banyak flamingo pada kandang dengan interval L hingga R secara inklusif.

Output

Anda dapat menanyakan maksimum N buah pertanyaan. Pertanyaan diawali dengan karakter “? L R“, ($1 \leq L < R \leq N$).

Untuk menebak jumlah flamingo, awali dengan dengan karakter “!“, diikuti dengan N buah bilangan bulat, dengan bilangan bulat ke- i menyatakan jumlah flamingo pada kandang ke- i . Setelah memberikan jawaban, program anda harus “exit“ atau dapat mendapatkan *verdict* “Idleness Limit Exceeded“.

Examples

standard input	standard output
6	? 1 2
5	? 5 6
15	? 3 4
10	! 1 4 4 6 7 8

Note

Misalkan jumlah flamingo dalam kandang-kandang misterius yang dimiliki oleh juri ialah $[1, 4, 4, 6, 7, 8]$.

Problem G. Gerak Lincah Maling Desa

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **256 megabytes**

Pak Chanek adalah seorang polisi di desa. Desa Pak Chanek dapat dinyatakan dengan sebuah grid G berukuran $N \times M$ dimana G_{ij} dapat berisi "." atau "#" yang secara berturut-turut menyatakan daerah yang dapat dilewati atau daerah terlarang (pagar, rumah orang, dan lain-lain). Pinggir-pinggir grid dianggap daerah terlarang yang tidak dapat dilewati oleh orang-orang.

Suatu hari, seorang maling membuat gaduh desa ini dan Pak Chanek berusaha menangkapnya. Baik maling dan Pak Chanek dalam satu detik dapat bergerak kedelapan arah sebanyak 1 petak (atas, bawah, kiri, kanan, kiri-atas, kiri-bawah, kanan-atas, kanan-bawah) dengan waktu pergerakannya dianggap instan. Sebagai contoh, anggap sekeliling Pak Chanek saat ini sebagai berikut:

```
 .##  
#C.  
...
```

Maka Pak Chanek dapat bergerak ke 5 arah yaitu: kiri-atas, kiri-bawah, bawah, kanan-bawah, dan kanan. Berikut skenario pengejaran setiap detiknya:

1. Tepat pada detik i , maling dapat bergerak 1 petak
2. Pada detik $(i + 0.5)$, Pak Chanek dapat bergerak 1 petak.

Pak Chanek maupun Maling dapat memilih untuk tidak bergerak pada suatu detik.

Jika Maling dan Pak Chanek bergerak secara optimal, tentukan apakah Pak Chanek akan dapat menangkap Maling, atau apakah Maling dapat selamanya menghindari dari Pak Chanek. Maling dikatakan tertangkap oleh Pak Chanek jika ada suatu waktu dimana posisi Maling dan Pak Chanek berhimpitan.

Input

Baris pertama berisi dua buah bilangan bulat N dan M ($1 \leq N, M \leq 10^3$), banyaknya baris dan banyaknya kolom pada grid secara berturut-turut.

N baris berikutnya masing-masing berisi M karakter. Karakter ke- j pada baris ke- i menyatakan G_{ij} ($G_{ij} \in \{ "C", "M", ".", "#" \}$). Akan terdapat tepat 1 karakter 'C' dan 'M' pada G . 'C' menyatakan posisi awal Pak Chanek dan 'M' menyatakan posisi awal maling.

Dijamin terdapat cara untuk mengunjungi setiap pasang petak non-"#" pada G .

Output

Keluarkan "tertangkap" tanpa tanda kutip jika Pak Chanek dapat menangkap maling. Keluarkan "lolos" tanpa tanda kutip jika maling dapat menghindari selamanya dari Pak Chanek.

Examples

standard input	standard output
5 5 ..#.# ##C. M....#..	tertangkap
4 7 ##.M.## ##.#.## ##.C.## #####	lolos

Note

Kemungkinan-kemungkinan pergerakan pada contoh masukan pertama:

- Jika maling bergerak ke atas:
 1. Pak Chanek bergerak kiri-bawah.
 2. Maling harus bergerak ke bawah jika tidak mau terperangkap
 3. Pak Chanek bergerak ke kiri.
 4. Jika maling bergerak ke atas atau ke bawah, pada giliran berikutnya akan ditangkap oleh Pak Chanek.
- Jika maling bergerak ke bawah:
 1. Pak Chanek bergerak ke kiri-bawah.
 2. Jika maling bergerak ke atas, kiri-atas, atau kanan, pasti akan ditangkap oleh Pak Chanek (kasus pertama). Jadi pilihan Maling adalah bergerak ke bawah, atau kanan-bawah.
 3. Baik ke kanan atau kanan-bawah, pada putaran berikutnya Pak Chanek bergerak ke kiri-bawah dan Maling akan ditangkap pada detik berikutnya.

Dapat ditunjukkan bahwa tidak ada kemungkinan untuk maling dapat menghindari Pak Chanek selamanya.

Untuk Contoh masukan kedua, maling dapat menghindari Pak Chanek selamanya dengan bergerak ke arah yang akan memaksimalkan jarak antara maling dan Pak Chanek.

Problem H. Hewan Mainan Chaneka

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **256 megabytes**

Chaneka mempunyai hobi main-mainan hewan. Setiap mainan-mainan Chaneka memiliki nilai keseruan sebuah bilangan real. Chaneka memiliki 4 kotak penyimpanan mainan yang dapat menyimpan mainan sebagai berikut:

- Kotak pertama menyimpan mainan dengan keseruan di $(\infty, -1]$
- Kotak kedua menyimpan mainan dengan keseruan di $(-1, 0)$
- Kotak ketiga menyimpan mainan dengan keseruan di $(0, 1)$
- Kotak keempat menyimpan mainan dengan keseruan di $[1, \infty)$.

Chaneka mempunyai A , B , C , dan D di kotak pertama, kedua, ketiga, dan keempat secara berturut-turut. Suatu hari Dia memutuskan hanya ingin satu mainan, sebuah mainan super. Chaneka kemudian membuat mainan super ini dengan menjahit mainan-mainan yang dia miliki.

Selama banyak mainan Chaneka masih lebih dari 1, Dia mengambil 2 mainan secara acak kemudian menjahitnya menjadi sebuah mainan baru. Nilai keseruan dari mainan ini adalah hasil perkalian dari nilai keseruan kedua mainan sebelum dijahit. Chaneka kemudian meletakkan mainan baru ini ke kotak yang sesuai. Dia terus melakukan ini sampai hanya tersisa total 1 mainan. Mainan terakhir inilah mainan super dan kotak yang berisi mainan super ini ditandai sebagai kotak spesial.

Sebagai orang luar, kamu hanya mengetahui jumlah mainan pada setiap kotak mula-mula namun tidak mengetahui nilai keseruan dari masing-masing mainan peserta. Kamu juga tidak tahu kronologi penjahitan Chaneka. Tentukan kotak mana saja yang mungkin menjadi kotak special setelah Chaneka menemukan mainan supernya.

Input

Baris pertama berisi sebuah bilangan bulat T ($1 \leq T \leq 5 \cdot 10^4$) yang merupakan banyaknya kasus.

Setiap kasus berisi sebuah baris berisi empat bilangan bulat A B C D ($0 \leq A, B, C, D \leq 10^6, A + B + C + D > 0$) dipisahkan oleh spasi yang menyatakan jumlah mainan pada kotak pertama, kedua, ketiga, dan keempat secara berurutan-turut.

Output

Untuk setiap kasus, keluarkan sebuah baris berisi empat string dipisahkan oleh spasi. Masing-masing string menyatakan apakah kotak pertama, kedua, ketiga, dan keempat mungkin menjadi kotak spesial, berurutan dari kiri ke kanan.

Untuk masing-masing kotak, cetak “Ya” apabila kotak tersebut mungkin menjadi kotak spesial atau cetak “Tidak” jika tidak.

Example

standard input	standard output
2	Ya Ya Tidak Tidak
1 2 0 1	Tidak Ya Tidak Tidak
0 1 0 0	

Note

Pada kasus pertama, berikut salah satu skenario dimana kotak pertama menjadi kotak spesial:

- Kotak pertama berisi mainan dengan keseruan $\{-3\}$
- Kotak kedua berisi mainan dengan keseruan $\{-0.5, -0.5\}$
- Kotak ketiga berisi mainan dengan keseruan $\{3\}$

Kronologi penjahitan:

1. Chaneka menjahit mainan dengan keseruan -0.5 dan -0.5 menjadi mainan dengan keseruan 0.25 lalu meletakkannya di kotak ketiga.
2. Chaneka menjahit mainan dengan keseruan -3 dan 0.25 menjadi mainan dengan keseruan -0.75 lalu meletakkannya di kotak kedua
3. Chaneka menjahit mainan dengan keseruan -0.75 dan 3 menjadi mainan dengan keseruan -1.25 lalu meletakkannya di kotak pertama yang kemudian menjadi kotak spesial.

Berikut salah satu skenario dimana kotak kedua menjadi kotak spesial:

- Kotak pertama berisi mainan dengan keseruan $\{-3\}$
- Kotak kedua berisi mainan dengan keseruan $\{-0.33, -0.25\}$
- Kotak ketiga berisi mainan dengan keseruan $\{3\}$

Kronologi penjahitan:

1. Chaneka menjahit mainan dengan keseruan -3 dan -0.33 menjadi mainan dengan keseruan 0.99 dan meletakkannya di kotak ketiga.
2. Chaneka menjahit mainan dengan keseruan 0.99 dan 3 menjadi mainan dengan keseruan 2.97 dan meletakkannya di kotak keempat.
3. Chaneka menjahit mainan dengan keseruan 2.97 dan -0.25 menjadi mainan dengan keseruan -0.7425 dan meletakkannya di kotak kedua yang kemudian menjadi kotak spesial.

Pada kasus kedua hanya terdapat 1 mainan sehingga mainan tersebut langsung menjadi mainan spesial dan Chaneka tidak perlu menjahit apa-apa.

Problem I. Indahnya Memanen Kebun Buah

Input file: **standard input**
Output file: **standard output**
Time limit: **7 seconds**
Memory limit: **256 megabytes**

Pak Chanek memiliki sebuah kebun buah berstruktur *rooted ternary tree* dengan N verteks yang dinomori dari 1 sampai N . *root* dari *tree* ini di verteks 1. P_i menyatakan *parent* dari verteks i , untuk setiap ($2 \leq i \leq N$). Menariknya, tinggi dari *tree* ini tidak lebih dari 10. Tinggi dari *tree* didefinisikan sebagai jarak terjauh dari *root* ke suatu verteks lain di *tree*.

Terdapat sebuah semak pada setiap verteks di kebun tersebut. Pada awalnya setiap semak di kebun Pak Chanek sedang berbuah. Buah tidak akan tumbuh pada suatu semak yang sudah memiliki buah. Semak pada verteks i , untuk setiap ($1 \leq i \leq N$) akan tumbuh buah A_i hari setelah buahnya terakhir dipanen.

Pak Chanek akan mengunjungi kebunnya selama Q hari. Pada hari ke- i , ia memanen seluruh semak yang sedang berbuah pada *subtree* verteks X_i di kebunnya, untuk setiap ($1 \leq i \leq Q$). Untuk setiap hari, tentukan jumlah jarak setiap semak yang dipanen ke X_i , serta berapa semak yang dipanennya pada hari itu.

Sebagai contoh, jika Pak Chanek memanen semua buah di subtree vertex X dan memanen semak $[Y_1, Y_2, \dots, Y_M]$, jumlah jarak setiap semaknya adalah $\sum_{i=1}^M \text{jarak}(X, Y_i)$.

$\text{jarak}(U, V)$ pada *tree* didefinisikan sebagai banyaknya *edge* pada *simple path* dari U ke V .

Input

Baris pertama berisi dua buah bilangan bulat, N dan Q ($1 \leq N, Q, \leq 5 \cdot 10^4$) yang berturut-turut menyatakan jumlah semak di kebun Pak Chanek dan jumlah hari Pak Chanek akan berjalan-jalan.

Baris kedua berisi N buah bilangan bulat A_i ($1 \leq A_i \leq 5 \cdot 10^4$), yang menyatakan kecepatan pertumbuhan semak pada verteks i di kebun Pak Chanek.

Baris ketiga berisi $N - 1$ buah bilangan bulat P_i ($1 \leq P_i \leq N, P_i \neq i$) untuk setiap ($2 \leq i \leq N$), yang menyatakan *parent* dari verteks ke i . Dijamin setiap verteks hanya dapat menjadi *parent* dari maksimal 3 verteks lain. Dijamin juga tinggi dari *tree* tidak lebih dari 10.

Q baris selanjutnya berisi sebuah bilangan bulat X_i ($1 \leq X_i \leq N$), untuk setiap ($1 \leq i \leq Q$), yang menyatakan tempat Pak Chanek memulai jalan-jalannya pada hari ke- i sesuai deskripsi.

Output

Q buah baris berisi dua buah bilangan, yaitu jumlah jarak setiap semak yang di panen ke X_i , serta jumlah semak yang dipanen Pak Chanek pada hari itu.

Examples

standard input	standard output
2 3 1 2 1 2 1 1	0 1 0 1 1 2
5 3 2 1 1 3 2 1 2 2 1 1 1 1	6 5 3 2 4 4

Note

Pada contoh masukan pertama:

- Pada hari pertama Pak Chanek mulai dari semak ke-2, dan hanya dapat memanen semak ke-2.
- Pada hari kedua Pak Chanek mulai dari semak ke-1, dan hanya dapat memanen semak ke-1, buah semak ke-2 belum dapat dipanen karena baru dipanen pada hari ke-1.
- Pada hari ketiga, Pak Chanek dapat memanen semak ke-1 dan 2. Jumlah jarak semua semak yang dipanen ialah 1.

Pada contoh masukan kedua, Pak Chanek selalu mulai dari semak ke-1. Pada hari pertama, kedua, dan ketika Pak Chanek berturut-turut memanen semak $[1, 2, 3, 4, 5]$, $[2, 3]$, $[1, 2, 3, 5]$.