



Problem A

Lama Menonton

Jingga suka menghabiskan waktu luangnya dengan menonton berbagai film, terutama film berseri (contoh: *k-drama*, *j-drama*, dan sebagainya). Jingga terbiasa menonton film-film tersebut melalui berbagai situs *streaming* seperti YouTube atau DailyMotion. Namun karena kecepatan internet yang dimiliki Jingga lambat, Jingga harus menunggu selama Y menit untuk proses *buffer* sebelum bisa menonton selama X menit (hal ini terus berulang selama durasi film tersebut). Proses *buffer* ini tidak terjadi pada X menit pertama; dengan kata lain, ketika Jingga menekan tombol *play*, maka ia bisa langsung menonton X menit pertama pada film yang ia tonton. Selain itu, Y menit waktu tunggu ini selalu terjadi meskipun sisa durasi film yang ia tonton kurang dari X menit.

Setelah menonton beberapa lama, Jingga kesal dan ingin mengetahui jam berapa dia akan selesai menonton film yang sedang ditonton apabila total durasi film tersebut adalah Z menit, dan setiap menonton selama X menit, Jingga harus menunggu Y menit untuk proses *buffer* sebelum bisa menonton kelanjutannya. Bantulah Jingga untuk menjawab pertanyaan tersebut.

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya kasus uji. Setiap kasus memiliki format $HH:MM \ X \ Y \ Z$ dan diberikan dalam satu baris. HH dan MM merepresentasikan jam dan menit ketika Jingga mulai menonton, sedangkan X , Y , dan Z ($1 \leq X \leq 3,600$; $0 \leq Y \leq 3,600$; $1 \leq Z \leq 12,000$) sesuai dengan deskripsi soal di atas. Format jam yang digunakan adalah format 24-jam (00:00 .. 23:59).

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: AA:BB" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan AA:BB menyatakan jam ketika Jingga menyelesaikan tontonannya dalam format 24-jam.

Contoh input	Output untuk contoh input
2 23:59 2 2 11 23:59 1 3 2	Kasus #1: 00:20 Kasus #2: 00:04



BNPC

The 2016
Bina Nusantara Programming Contest
for High School Students



- halaman ini sengaja dikosongkan -



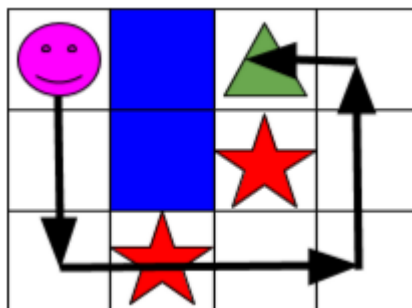
Problem B The Asterisk War

"I'll etch it powerfully with my one and only voice: This is The Asterisk War!"
- Shiena Nishizawa

Perang Asterisk sudah dimulai! Perang ini dilakukan di dalam sebuah *grid* berukuran $N \times M$ dan Sitomo sebagai ksatria yang baik hati ingin segera meninggalkan perang ini dengan meminimalkan pertumpahan darah yang akan terjadi.

Saat ini Sitomo berada pada sel bertanda 'Y', dan ia dapat meninggalkan perang ini dengan menuju sel yang bertanda 'T'. Pada sel bertanda 'T' tersebut tersedia *Transportal* yang merupakan sejenis alat untuk teleportasi. Selain itu, terdapat sel bertanda '#' yang merupakan *Barrier* yang tidak dapat dilalui oleh Sitomo. Sel yang bisa dilewati oleh Sitomo bertanda '.' (karakter titik) yang merupakan jalan kosong.

Musuh-musuh Sitomo, Asterisk, berada pada sel yang bertanda '*' (karakter bintang/asterisk). Apabila Sitomo melewati sel berisi Asterisk maka akan terjadi pertumpahan darah antara Sitomo dan Asterisk. Sitomo hanya dapat bergerak menuju sel-sel yang bersebelahan dengan sel yang ia tempati sekarang. Sehingga, apabila Sitomo sedang berada pada sel (X,Y) maka Sitomo hanya dapat bergerak ke kiri (X-1, Y), ke kanan (X+1, Y), ke atas (X, Y+1), dan ke bawah (X, Y-1) selama sel yang dituju bukanlah *Barrier* dan tidak keluar dari *grid*.



Sitomo membutuhkan bantuanmu untuk menghitung berapa banyak pertumpahan darah minimal yang harus dilakukan agar Sitomo dapat meninggalkan perang tersebut melalui *Transportal*. Sitomo adalah seorang ksatria yang kuat dan tidak akan pernah kalah apabila dia berhadapan dengan Asterisk. Setiap peta yang diberikan dijamin hanya memiliki sebuah *Transportal*.

Input

Baris pertama input dimulai dengan sebuah bilangan T ($T \leq 100$) yang menyatakan banyaknya kasus uji. Setiap kasus dimulai dengan dua buah bilangan N dan M ($2 \leq N, M \leq 300$) yang menandakan lebar dan panjang dari *grid* yang diinput. N baris selanjutnya masing-masing berisi M buah karakter yang merepresentasikan setiap sel yang ada dalam *grid* tersebut. Setiap sel hanya direpresentasikan oleh karakter 'Y', 'T', '#', '.', atau '*'.



BNPC

The 2016
Bina Nusantara Programming Contest
for High School Students



Output

Untuk setiap kasus, output “Kasus #X: Y” (tanpa kutip) di mana X adalah nomor kasus (dimulai dari 1 secara berurutan), dan Y adalah jumlah pertumpahan darah minimal yang harus dilakukan oleh Sitomo. Apabila Sitomo tidak dapat meninggalkan perang tersebut, keluarkan “-1” (tanpa tanda kutip) untuk Y.

Contoh input	Output untuk contoh input
1 3 4 Y#T. .#*. .*..	Kasus #1: 1

Penjelasan contoh kasus 1

Ini adalah contoh yang digunakan pada deskripsi soal di atas. Sitomo harus mengalami pertumpahan darah setidaknya satu kali.



Problem C

Jalan - Jalan

Jingga gemar sekali berjalan-jalan mengelilingi dunia dan ia telah melihat banyak sekali teknologi dan kebudayaan di berbagai negara.

Suatu ketika, ia diajak oleh Putra untuk berkunjung ke satu-satunya negara yang belum pernah dikunjungi oleh Jingga. Negara tersebut bernama Wakanda. Meskipun negara Wakanda ini tertutup dari peradaban dunia luar, tapi teknologi dan kebudayaan yang ada di negara ini selangkah lebih maju dari negara-negara lain.

Pada saat Jingga mengunjungi negara ini, ia menemukan suatu hal menarik, yaitu sebuah permainan puzzle yang bernama "Patching Numbers". Aturan permainannya adalah sebagai berikut:

1. Pemain diberikan sebuah bilangan bulat N , dan K buah bilangan bulat $A_{1..K}$ dengan jumlah $A_1 + A_2 + \dots + A_K$ berada di antara 1 dan N , inklusif.
2. Kemudian, pemain diminta untuk menambahkan nol atau lebih bilangan ke dalam K bilangan bulat tersebut ($A_{1..K}$ menjadi $A_{1..L}$ dengan $K \leq L$) sedemikian sehingga setiap bilangan dari 1 hingga N bisa dibentuk dengan menjumlahkan satu atau lebih bilangan yang ada pada $A_{1..L}$. Untuk membentuk sebuah bilangan, setiap bilangan yang ada pada $A_{1..L}$ hanya bisa digunakan tidak lebih dari satu kali. Dua bilangan yang nilainya sama adalah berbeda meskipun nilainya sama.

Contoh permainan yang berjalan adalah sebagai berikut :

- Jingga diberikan $N = 6$ dan $A_{1..2} = \{1, 3\}$.
- Bilangan yang bisa dibentuk dari $A_{1..2}$ adalah 1, 3, dan 4 (dari $1 + 3$). Agar semua bilangan dari 1 hingga 6 bisa dibentuk, maka Jingga harus menambahkan bilangan 2 ke dalam A , sehingga $A_{1..3} = \{1, 3, 2\}$. Dengan demikian, Jingga bisa membentuk: 1, 2, 3, 4 (dari $1 + 3$), 5 (dari $2 + 3$), dan 6 (dari $1 + 2 + 3$).

Karena Jingga adalah orang yang tidak mudah puas, dia tidak ingin berhenti sampai di situ. Ia ingin menyelesaikan setiap puzzle yang ada dengan menambahkan sedikit mungkin bilangan. Untuk setiap puzzle yang diberikan, bantulah Jingga untuk menentukan berapa minimum banyak bilangan yang harus ditambahkan ke dalam A .

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya puzzle yang harus dipecahkan. Baris pertama dari setiap puzzle berisi dua buah bilangan bulat N dan K ($1 \leq K < N \leq 10,000$). Baris berikutnya terdiri dari K buah bilangan bulat A_i ($1 \leq A_i \leq N$) yang merepresentasikan bilangan-bilangan pada $A_{1..K}$ dalam puzzle yang ingin dipecahkan Jingga.

Output

Untuk setiap puzzle yang diberikan, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y menyatakan minimum banyaknya bilangan yang harus ditambahkan Jingga agar ia dapat menyelesaikan puzzle tersebut.



BNPC

The 2016
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
3 6 2 1 3 20 3 1 5 10 5 3 1 2 2	Kasus #1: 1 Kasus #2: 2 Kasus #3: 0

Penjelasan contoh kasus 1

Contoh kasus ini adalah kasus yang digunakan pada deskripsi soal di atas. Jingga perlu menambahkan bilangan 2 sehingga $A_{1..3} = \{1, 3, 2\}$ dan ia bisa membentuk semua bilangan dari 1 hingga 6.

Penjelasan contoh kasus 2

Jingga perlu menambahkan bilangan 2 dan 4 sehingga $A_{1..5} = \{1, 5, 10, 2, 4\}$. Semua bilangan dari 1 hingga 20 bisa dibentuk dengan $A_{1..5}$:

- 1
- 2
- $3 \rightarrow 1 + 2$
- 4
- 5
- $6 \rightarrow 2 + 4$
- $7 \rightarrow 2 + 5$
- $8 \rightarrow 1 + 2 + 5$
- $9 \rightarrow 4 + 5$
- 10
- $11 \rightarrow 1 + 10$
- $12 \rightarrow 2 + 10$
- $13 \rightarrow 1 + 2 + 10$
- $14 \rightarrow 4 + 10$
- $15 \rightarrow 5 + 10$
- $16 \rightarrow 2 + 4 + 10$
- $17 \rightarrow 2 + 5 + 10$
- $18 \rightarrow 1 + 2 + 5 + 10$
- $19 \rightarrow 4 + 5 + 10$
- $20 \rightarrow 1 + 4 + 5 + 10$

Penjelasan contoh kasus 3

Jingga tidak perlu menambahkan bilangan apapun karena semua bilangan dari 1 hingga 5 bisa dibentuk dengan $A_{1..3} = \{1, 2, 2\}$ yang diberikan.



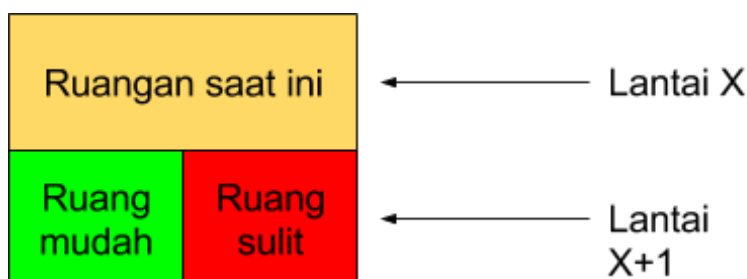
Problem D

Tree of Savage

Tree of Savage, game dengan genre MMORPG yang lebih sering dikenal dengan nama ToS, merupakan game MMORPG yang saat ini sedang populer baik di kalangan anak muda, remaja, hingga dewasa. Bahkan pembuat soal pun membuat soal ini saat sedang bermain ToS!

Kurang lebihnya game ini menggunakan sistem Dungeon. Suatu dungeon tersusun atas N buah lantai, dan kita memulai perjalanan kita dari lantai pertama, dan terus naik hingga lantai terakhir untuk menyelesaikan dungeon tersebut. Lantai ke- K memiliki 2^{K-1} buah ruangan, dengan kata lain lantai 1 memiliki 1 ($= 2^0$) buah ruangan, lantai 2 memiliki 2 ($= 2^1$) buah ruangan, lantai 3 memiliki 4 ($= 2^2$) buah ruangan, dan seterusnya. Setelah kita menyelesaikan suatu ruangan, kita mendapatkan 2 buah pilihan, menuju ruang mudah atau ruang sulit pada lantai selanjutnya. Setelah menyelesaikan lantai terakhir, barulah kita dapat keluar dari dungeon, dengan kata lain kita telah memenangkan dungeon tersebut.

Pada setiap ruangan, kita akan melawan Savage Beast, monster penjaga ruangan. Menyelesaikan suatu ruangan berarti berhasil mengalahkan Savage Beast. Setiap Savage Beast memiliki HP (*Hit Point*) dan dijamin HP dari semua Savage Beast pasti diantara L dan R (inklusif). Apabila kita memilih untuk menuju ruang mudah, maka HP dari Savage Beast pada semua lantai berikutnya akan lebih rendah dari HP Savage Beast pada lantai tersebut, begitu pula apabila kita memilih untuk menuju ruang sulit maka HP dari Savage Beast pada semua lantai berikutnya akan lebih tinggi dari HP Savage Beast pada lantai tersebut. Tentu apabila kita memilih ruang yang lebih sulit, hadiah yang kita dapatkan dari dungeon akan lebih besar, pastinya dengan risiko kalah yang lebih besar!



Cukup dengan penjelasan teknis dari ToS! Kini saatnya pembuat soal bermain! Sebelum masuk ke dungeon hendaknya kita menyusun strategi terlebih dahulu. Salah satu fitur menarik dari ToS adalah adanya Information Dealer. Tugas dari seorang Information Dealer adalah menjual peta dari suatu Dungeon dimana peta tersebut sudah termasuk HP dari Savage Beast yang ada. Tentunya pembuat soal sudah membeli peta tersebut. Sayangnya, agar game ini lebih menarik, developer dari ToS memutuskan untuk menyembunyikan informasi mengenai HP dari Savage Beast pada “beberapa” ruangan dari semua peta yang diperjualbelikan.

Karena bingung dengan beberapa ruangan yang informasinya tersembunyi, pembuat soal meminta bantuan anda untuk membuat program yang dapat mencari banyaknya kemungkinan susunan dungeon yang berbeda yang bisa dibentuk dari mengisi semua kemungkinan HP dari Savage Beast pada setiap ruangan yang masih tidak diketahui. Bantulah sang pembuat soal agar ia bisa bermain dengan tenang!



Input

Baris pertama input dimulai dengan sebuah bilangan T ($T \leq 100$) yang menyatakan banyaknya kasus uji. Setiap kasus dimulai dengan tiga buah bilangan bulat N , L , dan R ($1 \leq N \leq 6$, $1 \leq L \leq R \leq 200$) yang menyatakan banyaknya lantai dan rentang HP dari Savage Beast secara berurutan.

N baris berikutnya masing-masing berisi 2^{K-1} buah bilangan bulat A_{ij} ($L \leq A_{ij} \leq R$) yang menandakan HP dari Savage Beast di lantai ke- K ruangan ke- j (K diberikan secara berurutan dari 1 hingga N). A_{ij} yang bernilai -1 menandakan informasi ruangan yang belum diketahui.

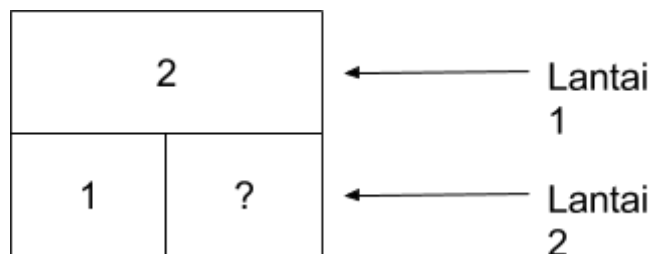
Output

Untuk setiap kasus, output "Kasus #X: Y" (tanpa kutip) di mana X adalah nomor kasus (dimulai dari 1 secara berurutan), dan Y adalah jawaban yang diminta setelah dimodulo dengan 1,000,000,007.

Contoh input	Output untuk contoh input
2 2 1 4 2 1 -1 1 1 200 -1	Kasus #1: 2 Kasus #2: 200

Penjelasan contoh kasus 1

Berikut adalah gambar dari contoh kasus pertama:



Pada contoh input pertama, diberikan lantai pertama ruang pertama memiliki Savage Beast dengan HP sebesar 2. Kemudian pada lantai kedua, ruang pertama (yang merupakan ruang mudah) memiliki Savage Beast dengan HP sebesar 1. Sedangkan ruang kedua pada lantai 2 merupakan ruang yang tidak diketahui. Karena ruangan tersebut merupakan ruang sulit, dengan batasan HP dari 1 hingga 4, maka akan terdapat 2 kemungkinan yang berbeda, yaitu HP Savage Beast sebesar 3 dan 4.

Penjelasan contoh kasus 2

Dungeon pada kasus ini hanya memiliki 1 lantai dengan 1 ruangan dan ruangan tersebut tidak diketahui informasinya. Karena batasan HP nya dari 1 hingga 200, maka terdapat 200 kemungkinan yang berbeda.



Problem E

Perkalian Sultan

Kriptografi adalah cabang ilmu komputer yang mempelajari penyajian data dalam format yang tersamarkan. Hal ini berguna ketika data yang dikirim lewat jalur publik tidak boleh dibaca oleh pihak yang tidak berhak. Kriptografi banyak dipakai untuk transmisi data penting (seperti perbankan) atau penyimpanan data penting dalam database (misal dalam menyimpan password yang telah di-hash).

Enkripsi adalah kegiatan menyamarkan suatu data dengan algoritma tertentu. Tingkat keamanan suatu algoritma enkripsi umumnya bergantung pada suatu properti matematika. Contoh properti matematika yang umum digunakan dalam enkripsi adalah sulitnya mencari faktor prima dari suatu bilangan, dengan adalah hasil perkalian 2 buah bilangan prima yang sangat besar sekali (biasanya mencapai 100 digit).

Pada suatu hari, Sultan merumuskan suatu aturan perkalian baru yaitu "**perkalian sultan**" (dengan simbol operator \diamond). Perkalian sultan untuk 2 buah bilangan bulat A dan B secara matematis dapat ditulis sebagai berikut:

$$A \diamond B = \sum_{i=1}^m \sum_{j=1}^n (a_i \times b_j)$$

di mana a_i adalah digit ke- i dari A, dan b_j adalah digit ke- j dari B. Dengan kata lain, perkalian sultan dari 2 buah bilangan bulat adalah jumlah hasil perkalian digit-digit penyusun bilangan A dengan digit-digit penyusun bilangan B. Perhatikan contoh perkalian sultan antara 15 dan 481 berikut sebagai contoh:

$$15 \diamond 471 = (1 \times 4) + (1 \times 7) + (1 \times 1) + (5 \times 4) + (5 \times 7) + (5 \times 1) = 4 + 7 + 1 + 20 + 35 + 5 = 72$$

Sultan mengamati bahwa jika diberikan dua buah bilangan bulat A dan B, kita dapat dengan mudah mencari hasil perkalian sultan kedua bilangan tersebut (misalkan $A \diamond B = C$). Namun jika diberikan bilangan hasil perkalian sultan C, sulit sekali mencari pasangan bilangan A dan B awal yang menghasilkan C (karena ada banyak sekali pasangan bilangan yang bisa menghasilkan C).

Sultan meyakini bahwa properti matematika ini dapat dimanfaatkan untuk mengembangkan algoritma enkripsi yang kuat. Tapi sebelum mengusulkannya ke organisasi ilmuwan kriptografi, Sultan meminta bantuan Anda untuk menguji properti matematika ini. Menurut Sultan, properti ini akan semakin kuat jika semakin banyak pasangan bilangan bulat, yang jika diaplikasikan ke dalam operasi perkalian sultan, akan menghasilkan bilangan yang sama.

Sultan akan memberikan Anda sebuah bilangan C. Anda diminta untuk mencari berapa banyak pasangan bilangan bulat (A, B) berbeda sedemikian sehingga $A \diamond B = C$. Karena Sultan tidak suka dengan bilangan 0 (dan juga karena bilangan yang mengandung digit 0 akan mengacaukan sistem perkalian sultan), Anda hanya perlu mencari pasangan bilangan di mana kedua bilangan tidak mengandung digit 0. Dua buah pasangan bilangan bulat (A, B) dan (A', B') dianggap berbeda jika $A \neq A'$ atau $B \neq B'$.



BNPC

The 2016
Bina Nusantara Programming Contest
for High School Students



Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya kasus yang diberikan oleh Sultan. Setiap kasus terdiri dari sebuah bilangan bulat C ($1 \leq C \leq 10,000,000$) dalam satu baris yang menyatakan hasil perkalian sultan.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y menyatakan banyaknya pasangan bilangan bulat (A, B) sedemikian sehingga $A \diamond B = C$, dan baik A maupun B tidak memiliki digit penyusun 0. Karena hasilnya bisa sangat besar, cetak Y setelah dimodulo dengan 1,000,000,007.

Contoh input	Output untuk contoh input
2 3 4	Kasus #1: 8 Kasus #2: 20

Penjelasan contoh kasus 1

Hanya terdapat 8 pasang bilangan bulat yang tidak memiliki digit 0 dan hasil perkalian sultan mereka adalah 3: (1, 3), (1, 12), (1, 21), (1, 111), (3, 1), (12, 1), (21, 1), dan (111, 1).



Problem F Utak Atik Lampu

Sitomo adalah orang yang sangat jahil; dia memiliki hobi menyalakan dan mematikan lampu untuk tujuan yang tidak kita pahami. Rumah Sitomo memiliki N lampu dan masing-masing lampu dikendalikan oleh sebuah saklar tekan (untuk menyalakan atau mematikan lampu, Sitomo cukup menekan saklar lampu yang bersangkutan).

Semua lampu di rumah Sitomo diberi nomor (ini adalah hal lain yang tidak kita pahami dari Sitomo) dari 1 hingga N . Kemudian Sitomo melakukan rutinitas yang tergolong aneh ini:

- Pada awalnya, semua lampu dalam keadaan mati.
- Pada pagi hari ke- K (dimulai dari 1), Sitomo akan menekan semua saklar yang nomor lampunya adalah kelipatan dari K dan membiarkan lampu-lampu tersebut demikian hingga hari berikutnya. Hal ini dilakukan terus setiap hari hingga N hari berlalu.

Pada hari pertama, semua lampu akan menyala (Sitomo menekan semua saklar). Pada pagi hari kedua, Sitomo menekan semua saklar yang nomor lampunya adalah kelipatan 2, ct: 2, 4, 6, 8, Karena lampu-lampu tersebut dalam keadaan menyala, maka penekanan ini akan menyebabkan lampu-lampu tersebut menjadi mati. Sehingga pada hari kedua, lampu yang menyala adalah: 1, 3, 5, 7, Pada pagi hari ketiga, Sitomo menekan semua saklar yang nomor lampunya adalah kelipatan 3, ct: 3, 6, Lampu ke-3 dalam keadaan menyala, sehingga setelah penekanan saklar, lampu ini akan mati; sedangkan, lampu ke-6 dalam keadaan mati, sehingga setelah penekanan saklar, lampu ini akan menyala. Dengan demikian pada hari ketiga, lampu yang menyala adalah: 1, 5, 6, 7,

Teman baik Sitomo, Jingga, penasaran dengan efek rutinitas Sitomo terhadap lampu-lampu di rumah Sitomo ini. Khususnya, Jingga ingin mengetahui ada berapa lampu dalam keadaan menyala di rumah Sitomo jika ia datang pada sore hari ke- X . Bantu Jingga.

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya kasus. Setiap kasus terdiri dari dua bilangan bulat N dan X ($1 \leq X \leq N \leq 1,000$) yang menyatakan banyaknya lampu di rumah Sitomo dan hari yang ingin diketahui oleh Jingga secara berurutan.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus # X : Y " (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y adalah jumlah lampu yang menyala pada hari yang ditanyakan.

Contoh input	Output untuk contoh input
2 7 1 8 2	Kasus #1: 7 Kasus #2: 4

Penjelasan contoh kasus 1

Lampu yang menyala pada sore hari ke-1: 1, 2, 3, 4, 5, 6, 7.

Penjelasan contoh kasus 2

Lampu yang menyala pada sore hari ke-2: 1, 3, 5, 7.



BNPC **The 2016**
Bina Nusantara Programming Contest
for High School Students



- halaman ini sengaja dikosongkan -



Problem G

Rivalitas

Phi dan Andri adalah teman dari masa kecil. Mereka sangat menyukai kompetisi. Di mana dan kapan pun mereka berada, mereka selalu berkompetisi. Mulai dari main, pelajaran, cari jodoh, bahkan hingga porsi makan.

Suatu hari, mereka berdua diajak oleh Bibi, teman mereka, untuk makan bersama. Dalam acara jamuan makan bersama ini, ada N jenis makanan. Andri dan Phi memiliki kebiasaan yang cukup unik, yaitu tidak ingin makan makanan yang sama untuk kedua kalinya dalam acara makan-makan. Karena Phi sedang diet, dia ingin makan dalam jumlah sedikit. Akan tetapi, karena rivalitasnya terhadap Andri, ia memutuskan untuk tidak ingin makan terlalu sedikit dari jumlah jenis makanan yang dimakan Andri, lebih tepatnya, ia memutuskan untuk makan tepat satu jenis makanan lebih sedikit dari Andri.

Melihat rivalitas Andri dan Phi, Bibi memiliki ide untuk membuat sayembara bahwa siapa pun yang dapat dengan tepat menghitung banyak kemungkinan menu yang Phi dan Andri makan, akan ditraktir makan. Teman-teman Bibi pun tertarik dengan sayembara tersebut. Mulai dari Cici hingga Zizi menerima tantangan tersebut untuk menghitung banyak kemungkinan menu yang Phi dan Andri makan.

Melihat banyaknya peserta yang tertarik akan sayembara tersebut, Bibi pun harus segera mengetahui jawaban yang tepat (banyaknya kemungkinan menu yang Phi dan Andri makan). Namun, Bibi pun tidak sanggup untuk menghitungnya; oleh karena itu, Bibi pun meminta bantuan Anda untuk menghitung jawabannya.

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya kasus. Setiap kasus terdiri dari sebuah bilangan bulat N ($1 \leq N \leq 1,000,000$) yang menyatakan banyaknya jenis makanan yang tersedia pada jamuan makan tersebut.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y menyatakan banyaknya kemungkinan menu yang disantap oleh Phi dan Andri. Karena hasilnya bisa sangat besar, cetak Y setelah dimodulo dengan 1,000,000,007.

Contoh input	Output untuk contoh input
1 3	Kasus #1: 15



Penjelasan contoh kasus 1

Ada 3 jenis makanan pada acara ini; kita asumsikan ketiga jenis makanan tersebut adalah: Nasi Goreng, Ketoprak, dan Soto Ayam. Berikut adalah semua kemungkinan menu yang bisa dipilih Andri dan Phi:

#	Andri			Phi		
	Nasi Goreng	Ketoprak	Soto Ayam	Nasi Goreng	Ketoprak	Soto Ayam
1	ya	-	-	-	-	-
2	-	ya	-	-	-	-
3	-	-	ya	-	-	-
4	ya	ya	-	ya	-	-
5	ya	ya	-	-	ya	-
6	ya	ya	-	-	-	ya
7	ya	-	ya	ya	-	-
8	ya	-	ya	-	ya	-
9	ya	-	ya	-	-	ya
10	-	ya	ya	ya	-	-
11	-	ya	ya	-	ya	-
12	-	ya	ya	-	-	ya
13	ya	ya	ya	ya	ya	-
14	ya	ya	ya	ya	-	ya
15	ya	ya	ya	-	ya	ya

Pada kemungkinan menu #1, Andri memilih Nasi Goreng, sedangkan Phi tidak makan. Pada kemungkinan menu #8, Andri memilih Nasi Goreng dan Soto Ayam, sedangkan Phi memilih Ketoprak. Pada kemungkinan menu #15, Andri memilih ketiga jenis makanan, sedangkan Phi memilih Ketoprak dan Soto Ayam.

Perhatikan, pada setiap kemungkinan menu, Phi memilih tepat satu jenis makanan lebih sedikit dari Andri.



Problem H

Lucky Sequence

Tahukah anda tentang *game show* di televisi yang bernama Lucky Draw? Permainannya sederhana. Terdapat N buah kotak, masing masing kotak diberi nomor terurut dari 1, 2, 3, ..., N . Kemudian anda memilih 1 buah bilangan diantara 1 dan N , dan anda akan mendapatkan hadiah yang terdapat pada kotak dengan nomor yang anda pilih! Sayangnya, kotak tersebut belum tentu berisi barang yang berguna, di mana dalam permainan ini kotak tersebut diberi nama Zonk.

Saat ini, Lanea sudah bosan dengan permainan Lucky Draw biasa, maka ia membuat permainan baru yang ia beri nama Lucky Sequence. Sama seperti Lucky Draw, diberikan N buah kotak dan masing masing kotak diberi nomor terurut dari 1, 2, 3, ..., N . Tentunya diantara kotak kotak tersebut terdapat kotak Zonk. Kemudian anda memilih 2 buah angka, L dan R , di antara 1 dan N ($L \leq R$) dan anda akan mendapatkan hadiah yang terdapat pada kotak dengan nomor di antara L dan R (inklusif)!

Sayangnya setelah beberapa game diadakan, ternyata (tentunya) semua orang hanya memilih angka $L = 1$ dan $R = N$, dengan kata lain dengan mudahnya mengambil semua barang yang ada dalam semua kotak. Menyadari sistem tersebut, Lanea membuat sistem yang baru, di mana kedua angka tersebut diambil secara acak, sehingga pemain hanya mengandalkan keberuntungannya untuk mendapatkan hadiah sebanyak-banyaknya.

Saat ini anda dipilih sebagai *beta tester* dari sistem yang baru! Anda sudah mengetahui semua barang yang tertera pada semua kotak, baik yang Zonk maupun yang tidak. Kemudian anda diminta untuk menghitung rata-rata banyak **kotak tidak Zonk** yang diambil pemain dari semua kemungkinan pengambilan yang ada. Karena jumlah kotak yang cukup banyak, maka anda memutuskan untuk membuat program yang dapat menghitung rata-rata yang diminta. Silakan diselesaikan.

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 100$) yang menyatakan banyaknya kasus. Setiap kasus dimulai dengan sebuah bilangan bulat N ($1 \leq N \leq 10,000$) yang menyatakan banyaknya kotak. Baris berikutnya berisi N buah bilangan bulat A_i ($A_i \in \{0, 1\}$) yang merepresentasikan isi kotak ke- i . $A_i = 1$ apabila kotak tersebut bukan Zonk, sebaliknya, $A_i = 0$ jika kotak tersebut adalah Zonk.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y adalah rata-rata banyak kotak Zonk pada permainan Lucky Sequence di kasus tersebut, dengan ketepatan 3 angka di belakang koma.

Contoh input	Output untuk contoh input
2 3 1 0 1 5 0 0 0 0 0	Kasus #1: 1.000 Kasus #2: 0.000



Penjelasan contoh kasus 1

Ada 6 kemungkinan nilai L dan R yang bisa dipilih dari 1..3.

- L = 1, R = 1 → kotak dengan nomor 1 hingga 1: 1 kotak tidak Zonk
- L = 1, R = 2 → kotak dengan nomor 1 hingga 2: 1 kotak tidak Zonk
- L = 1, R = 3 → kotak dengan nomor 1 hingga 3: 2 kotak tidak Zonk
- L = 2, R = 2 → kotak dengan nomor 2 hingga 2: 0 kotak tidak Zonk
- L = 2, R = 3 → kotak dengan nomor 2 hingga 3: 1 kotak tidak Zonk
- L = 3, R = 3 → kotak dengan nomor 3 hingga 3: 1 kotak tidak Zonk

Sehingga rata-ratanya adalah $(1 + 1 + 2 + 0 + 1 + 1) / 6 = 1.000$



Problem I

Membagi Permen

Vina adalah seorang wanita kemayu namun juga adil dan baik hatinya; dia memiliki harapan mulia agar semua orang berbahagia. Suatu hari, Vina membawa M permen, dan ingin membagikan semua permen tersebut **secara merata** kepada sejumlah orang. Setelah ia bagikan, ternyata tersisa sejumlah N permen. Vina begitu antusias menceritakan kisah permennya kepada anda, teman dekatnya. Akan tetapi, Vina lupa berapa banyak orang yang mendapat permen darinya, dan ia pun menjadi uring-uringan di depan Anda.

Anda yang sering terganggu dengan sikap centil Vina, ingin segera menyelesaikan masalah ini. Segera bantu Vina untuk mengingat dengan menghitung ada berapa banyak kemungkinan jumlah orang yang mendapatkan permen tersebut.

Input

Baris pertama berisi sebuah bilangan T ($1 \leq T \leq 1,000$) yang menyatakan banyaknya kasus yang harus diselesaikan. Setiap kasus terdiri dari dua buah bilangan bulat M dan N ($0 \leq N < M \leq 500,000$) yang menyatakan jumlah permen yang dibawa Vina dan jumlah permen yang tersisa secara berurutan.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip), dengan X menyatakan nomor kasus dimulai dari 1, dan Y adalah banyaknya kemungkinan jumlah orang yang mendapatkan permen pada kasus tersebut.

Contoh input	Output untuk contoh input
2 13 2 11 3	Kasus #1: 1 Kasus #2: 2

Penjelasan contoh kasus 1

Hanya ada satu kemungkinan jumlah orang yang mendapatkan permen, yaitu: 11. Dengan 11 orang dan 13 permen, jika masing-masing orang mendapatkan 1 permen, maka akan tersisa 2 permen.

Penjelasan contoh kasus 2

Di kasus ini, ada dua kemungkinan jumlah orang yang mendapatkan permen, yaitu: 4 dan 8. Dengan 4 orang, 11 permen, dan masing-masing orang mendapatkan 2 permen, permen yang tersisa adalah 3. Dengan 8 orang, 11 permen, dan masing-masing orang mendapatkan 1 permen, permen yang tersisa adalah 3.



BNPCHS

The 2016
Bina Nusantara Programming Contest
for High School Students

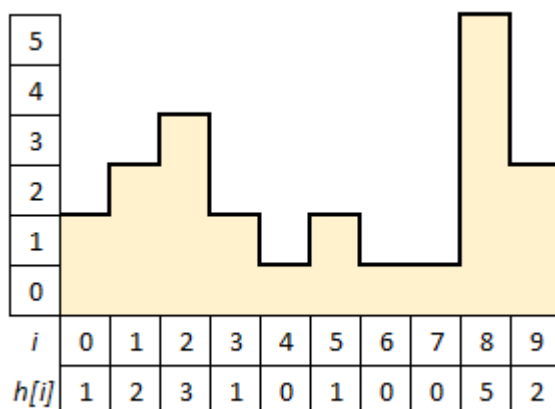


- halaman ini sengaja dikosongkan -



Problem J Proyek Tanah

Putra dan timnya yang bernama Proyek Monster berhasil mendapatkan proyek untuk membangun sebuah kampus. Putra memiliki lahan kosong dengan N petak yang berderetan dengan tinggi awal yang mungkin berbeda-beda.



Gambar di atas merupakan contoh jika petak yang diberikan adalah 10, dan tinggi awal dari masing masing tanah adalah [1, 2, 3, 1, 0, 1, 0, 0, 5, 2].

Putra ingin membuat lahan kosong yang ia miliki menjadi sama tinggi. Untuk hal ini, Putra memiliki N Bos Kecil, dan setiap Bos Kecil bertanggung jawab terhadap satu dari N petak lahan kosong tersebut. Bos Kecil yang bertanggung jawab terhadap petak tanah ke- i bisa menambahkan tinggi tanah pada petak ke- i sebesar tepat X_i . Hanya pada N Bos Kecil inilah Putra menggantungkan harapannya.

Biaya menambah tinggi tanah tentunya tidaklah murah; oleh karena itu, Putra memerlukan bantuan kalian untuk menentukan berapa tinggi minimal agar semua N petak tanah tersebut sama tingginya.

Input

Baris pertama berisi sebuah bilangan T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus diawali dengan sebuah bilangan bulat N ($2 \leq N \leq 1,000$) yang menyatakan banyaknya petak tanah pada lahan kosong milik Putra. Baris berikutnya berisi N buah bilangan bulat h_i ($0 \leq h_i \leq 1,000,000$) yang menyatakan tinggi awal dari masing-masing petak. Baris berikutnya berisi N buah bilangan bulat X_i ($1 \leq X_i \leq 1,000,000$) yang menyatakan tinggi tanah yang dapat ditambahkan pada petak ke- i .

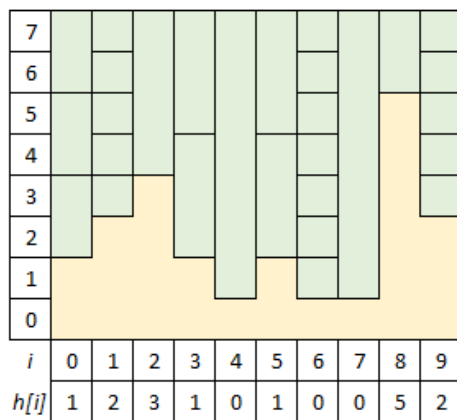
Output

Untuk setiap kasus, cetak "Kasus #X: Y". Dimana X adalah nomor kasus dimulai dari 1, dan Y adalah jawaban dari kasus tersebut. Dijamin jawaban pasti tersedia untuk setiap input dan tidak lebih dari 10^{18} .

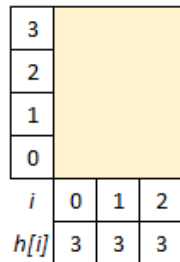


Contoh input	Output untuk contoh input
3 10 1 2 3 1 0 1 0 0 5 2 2 1 4 3 7 3 1 7 2 1 3 3 3 3 1 3 2 3 1 2 3 3 1 2	Kasus #1: 7 Kasus #2: 3 Kasus #3: 7

Penjelasan contoh kasus 1



Penjelasan contoh kasus 2



Penjelasan contoh kasus 3

