



Problem A

Lomba Makan Beregu

Lomba Makan Beregu (LMB) 2015 akan dilaksanakan kurang dari tiga bulan lagi. *Bina Nusantara Party Club* (BNPC), sebagai juara bertahan, tentu saja akan kembali mengikuti kejuaraan bergengsi yang satu ini.

LMB mirip dengan kompetisi pemrograman pada umumnya, namun dilakukan secara beregu (tim), dengan satu tim terdiri dari tepat 3 orang. Pada saat perlombaan, setiap tim akan diberikan 3 jenis makanan yang harus dihabiskan dalam waktu kurang dari satu jam. Tim yang berhasil menghabiskan paling banyak makanan akan keluar sebagai pemenang. Jenis makanan yang dilombakan selalu sama setiap tahunnya, yaitu:

- (1) Ayam Goreng : tentunya kalian tahu apa itu ayam yang digoreng.
- (2) Combro : makanan khas Jawa Barat; terbuat dari parutan singkong yang dibentuk bulat dan bagian dalamnya diisi dengan oncom, kemudian digoreng.
- (3) Martabak Telur : terbuat dari adonan tepung terigu dan adonan telur yang dicampur dengan daging cincang dan rempah, dan kemudian digoreng.

Dari pengalaman mereka mengikuti lomba serupa pada tahun-tahun sebelumnya, BNPC mengetahui bahwa, untuk menang, tidak cukup hanya dengan memakan satu atau dua dari tiga jenis makanan yang disediakan; mereka harus menghabiskan ketiganya.

BNPC saat ini memiliki N anggota aktif, dan masing-masing anggota ini memiliki keahlian khusus dalam memakan salah satu dari 3 jenis makanan di atas. BNPC tentunya ingin kembali menyabet seluruh penghargaan di LMB, oleh karenanya, mereka ingin mengirim sebanyak mungkin tim untuk mengikuti perlombaan ini. **Setiap anggota tentunya tidak boleh berada pada dua atau lebih tim yang berbeda, dan satu tim harus terdiri dari tepat 3 orang yang memiliki keahlian khusus pada makanan yang berbeda.** Berapa banyak tim yang bisa mereka ikut sertakan ke dalam perlombaan ini?

Contoh. Ada 6 anggota BNPC yang masing-masing memiliki keahlian memakan: ayam goreng, martabak telur, martabak telur, combro, ayam goreng, dan martabak telur. Dari 6 anggota ini, kita hanya bisa membentuk satu tim karena hanya ada satu anggota yang memiliki keahlian dalam memakan combro di antara 6 anggota tersebut.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan sebuah bilangan bulat N ($1 \leq N \leq 100$) yang menyatakan banyaknya anggota BNPC. Baris berikutnya berisi N buah bilangan bulat P_i ($P_i \in \{1, 2, 3\}$) yang merepresentasikan makanan keahlian dari setiap anggota. $P_i = 1$ artinya anggota ke- i memiliki keahlian dalam memakan ayam goreng, $P_i = 2$ artinya anggota tersebut memiliki keahlian dalam memakan combro, dan $P_i = 3$ untuk martabak telur.

Output

Untuk setiap kasus, output dalam satu baris "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah banyaknya tim yang bisa mereka ikut sertakan ke dalam LMB 2015.



BNPC

The 2015
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
4 6 1 3 3 2 1 3 4 2 2 3 2 7 1 2 3 1 2 3 1 10 3 3 3 2 1 2 3 2 1 1	Kasus #1: 1 Kasus #2: 0 Kasus #3: 2 Kasus #4: 3

Penjelasan contoh kasus 1

Hanya satu tim yang bisa dibentuk dari 6 orang yang ada, karena hanya ada 1 orang yang memiliki keahlian memakan combro (2). Kasus ini sama dengan contoh yang digunakan pada penjelasan soal di atas.

Penjelasan contoh kasus 2

Tidak ada tim yang bisa dibentuk, karena tidak ada satupun anggota yang memiliki keahlian dalam memakan ayam goreng.

Penjelasan contoh kasus 3

Kita bisa membentuk dua tim pada kasus ini dan akan tersisa satu anggota dengan keahlian memakan ayam goreng yang tidak akan mengikuti lomba tersebut.



Problem B Dunia Mimpi

Lusi sangat menyukai film *Inception*, yang dimainkan oleh aktor kesayangannya, *Leonardo DiCaprio*. Film ini menceritakan kisah Cobb (dimainkan oleh Leonardo DiCaprio) yang memiliki misi menyisipkan ide/gagasan ke alam bawah sadar seorang putra konglomerat, melalui mimpi. Agar misi penanaman gagasan ini bisa berhasil, Cobb harus masuk ke dalam mimpi seseorang beberapa tingkat lebih dalam. Mimpi tingkat 1 adalah mimpi ketika kita tidur. Mimpi tingkat 2 adalah mimpi dari mimpi kita (kita bermimpi kalau kita sedang bermimpi). Mimpi tingkat 3 adalah mimpi dari mimpi dari mimpi kita, dan seterusnya. Konsep mimpi bertingkat ini sering disalah artikan sebagai inception, padahal *inception* memiliki arti penanaman gagasan di film ini.

Dalam film *Inception*, waktu di mimpi tingkat ke A berjalan lebih lambat dari mimpi di tingkat ke A+1 (1 detik di mimpi tingkat ke A adalah B detik di mimpi tingkat ke A+1, di mana $B \geq 1$). A = 0 menyatakan dunia nyata. Namun, di dunia tempat Lusi berada, yang terjadi adalah sebaliknya. Waktu di mimpi tingkat ke A berjalan lebih cepat dari mimpi di tingkat ke A+1 (1 detik di mimpi tingkat ke A+1 adalah B detik di mimpi tingkat ke A, dengan $B \geq 1$). Contoh. Sebuah mimpi 3 tingkat yang memiliki pola waktu <2, 5, 4> memiliki arti:

- mimpi di dunia nyata berjalan 2 kali lebih cepat dari mimpi tingkat 1.
 - 1 detik di mimpi tingkat 1 = 2 detik di dunia nyata.
- mimpi di tingkat 1 berjalan 5 kali lebih cepat dari mimpi di tingkat 2.
 - 1 detik di mimpi tingkat 2 = 5 detik di mimpi tingkat 1.
 - 1 detik di mimpi tingkat 2 = $2 * 5$ detik di dunia nyata.
- mimpi di tingkat 2 berjalan 4 kali lebih cepat dari mimpi di tingkat 3.
 - 1 detik di mimpi tingkat 3 = 4 detik di mimpi tingkat 2.
 - 1 detik di mimpi tingkat 3 = $5 * 4$ detik di mimpi tingkat 1.
 - 1 detik di mimpi tingkat 3 = $2 * 5 * 4$ detik di dunia nyata.

Suatu malam Lusi bertemu dengan Jin Botak. Jin Botak memiliki kemampuan untuk masuk ke berbagai tingkat mimpi dalam dunia mimpi, dan ia bisa membawa Lusi besertanya. Jin Botak hanya akan memberikan kekuatan ini kepada Lusi jika Lusi bersedia mengerjakan beberapa pekerjaan dari Jin Botak di dunia mimpi.

Pekerjaan dari Jin Botak diberikan pada N tingkat dunia mimpi, di mana pekerjaan pada mimpi tingkat ke-i membutuhkan waktu selama P_i di mimpi tingkat tersebut. Lusi ingin mengetahui, berapa total waktu yang ia perlukan untuk mengerjakan semuanya dalam hitungan waktu di dunia nyata.

Contoh. Diberikan pola kecepatan waktu di ($N = 3$) tingkatan mimpi: $W_1 = 2$, $W_2 = 5$, $W_3 = 4$, dan pekerjaan dari Jin Botak di masing-masing tingkatan mimpi: $P_1 = 10$, $P_2 = 5$, $P_3 = 1$. Total waktu yang diperlukan Lusi untuk mengerjakan semua pekerjaan yang diberikan jika dihitung berdasarkan waktu pada tingkat ke 3 adalah:

- Pekerjaan di mimpi tingkat 1: $10 * 2 = 20$
- Pekerjaan di mimpi tingkat 2: $5 * (2 * 5) = 50$
- Pekerjaan di mimpi tingkat 3: $1 * (2 * 5 * 4) = 40$

Sehingga total waktu yang diperlukan Lusi dalam dunia nyata adalah $20 + 50 + 40 = 110$.



Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus anda tangani. Setiap kasus diawali dengan sebuah bilangan bulat N ($1 \leq N \leq 100$) yang menyatakan banyaknya tingkatan mimpi. Baris berikutnya berisi N buah bilangan bulat W_i ($1 \leq W_i \leq 10.000.000$) yang menyatakan bahwa 1 detik di mimpi tingkat ke i sama dengan W_i detik di mimpi tingkat ke $i - 1$ (1 tingkat di atasnya). Mimpi tingkat 0 adalah sama dengan dunia nyata. Baris berikutnya berisi N buah bilangan bulat P_i ($0 \leq P_i \leq 10.000.000$) yang menyatakan lama pekerjaan yang harus dikerjakan di mimpi tingkat ke i dalam waktu menurut mimpi tingkat ke i .

Output

Untuk setiap kasus, output dalam satu baris, "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah total waktu yang diperlukan Lusi untuk menyelesaikan seluruh tugas yang diberikan menurut waktu di dunia nyata. Nilai output dijamin tidak lebih besar dari 1.000.000.000.

Contoh input	Output untuk contoh input
3 2 5 4 5 4 3 2 3 4 3 2 1 2 2 5 2 2	Kasus #1: 105 Kasus #2: 42 Kasus #3: 24

Penjelasan untuk contoh input 1:

Pada mimpi tingkat 1, waktu berjalan 5 kali lebih lambat dari dunia nyata dan Lusi perlu melakukan pekerjaan selama 5 unit waktu (setara dengan 25 unit waktu di dunia nyata). Untuk pekerjaan tingkat kedua, Lusi memerlukan 4 unit waktu di mimpi tingkat 2, atau setara 16 unit waktu di mimpi tingkat 1 (yang 4 kali lebih cepat dari mimpi tingkat 2), atau setara 80 unit waktu di dunia nyata (yang 5 * 4 kali lebih cepat dari mimpi tingkat 2). Dengan demikian total waktu yang diperlukan Lusi untuk menyelesaikan kedua pekerjaan tersebut menurut hitungan waktu dunia nyata adalah $25 + 80 = 105$ unit waktu.

**BNPC**

The 2015
Bina Nusantara Programming Contest
for High School Students



Problem C

Makan Teman

Besok adalah hari pertama UTS (Ujian Tengah Semester) di Universitas Panda. Akan tetapi, Antonio dengan santainya mengajak N orang teman-temannya untuk memainkan permainan (komputer) terbaru bernama Makanio. Makanio adalah sebuah permainan di mana seorang pemain harus memakan lawan-lawannya agar menjadi semakin besar.

Aturan dalam game ini adalah:

- Pemain A dapat memakan pemain B apabila skor A lebih besar dari skor B.
- Apabila A memakan B, maka skor milik B akan ditambahkan ke A.
- Pemain yang telah dimakan akan dikeluarkan dari permainan.

Ricko adalah saingan Antonio dalam permainan ini dan dia tidak termasuk di antara N teman yang diajak bermain oleh Antonio. Pada malam sebelum UTS ini, Antonio, N teman Antonio, dan Ricko bermain dalam permainan Makanio ini; tidak ada pemain lain yang ikut bermain pada saat ini. Antonio sangat ingin mengalahkan (baca: memakan) Ricko di Makanio. Untuk mengalahkan Ricko, Antonio mungkin perlu memakan beberapa temannya sehingga skor Antonio menjadi lebih besar dari skor Ricko. Namun Antonio hanya ingin memakan sebanyak jumlah temannya yang diperlukan, lebih sedikit lebih baik, bahkan jika bisa ia tidak ingin memakan satu pun temannya (yang hanya bisa terjadi jika skor Antonio sudah lebih besar dari skor Ricko pada awalnya).

Untuk hal ini, Antonio meminta bantuan Gilberto untuk menghitung berapa jumlah minimum temannya yang perlu ia makan agar bisa memakan Ricko. Teman-teman Antonio sangat lihai dalam melarikan diri, sehingga mereka tidak akan bisa dimakan oleh Ricko. Namun teman-teman Antonio bersedia berkorban jika yang akan memakan mereka adalah Antonio.

Gilberto tentu saja sedang belajar untuk UTS! Dia dengan segera meminta bantuan anda menyelesaikan masalah Antonio. Sebagai siswa SMA yang belum kuliah, tentunya anda memiliki waktu luang yang lebih banyak (demikian pikiran Gilberto); bantu Antonio dan Gilberto.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya kasus yang harus anda tangani. Setiap kasus dimulai dengan satu baris dengan tiga buah bilangan bulat N , M , dan R ($1 \leq N \leq 50.000$; $1 \leq M \leq 10^9$; $1 \leq R \leq 2 \cdot 10^9$) yang menyatakan banyaknya teman Antonio, skor awal Antonio, dan skor Ricko secara berurutan. Baris berikutnya berisi N buah bilangan bulat A_i ($1 \leq A_i \leq 10^6$) yang menyatakan skor teman Antonio ke- i secara berurutan.

Output

Untuk setiap kasus, output dalam satu baris, "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah jumlah minimum teman Antonio yang harus dimakan oleh Antonio. Apabila tidak ada cara agar Antonio bisa memakan Ricko, keluarkan "UTS!" (tanpa kutip) sebagai pengganti Y.



BNPC

The 2015
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
4 3 10 15 1 5 10 5 10 20 1 2 3 4 5 5 100 200 20 40 40 50 145 7 15 1000 1 2 3 4 5 6 7	Kasus #1: 2 Kasus #2: 3 Kasus #3: 2 Kasus #4: UTS!

Penjelasan untuk contoh kasus 1:

Skor Antonio mula-mula adalah 10, sedangkan skor Ricko adalah 15. Antonio perlu memakan dua orang temannya: yang memiliki skor 5 dan skor 1. Setelah Antonio memakan dua orang tersebut, skor Antonio menjadi 16, dan siap memakan Ricko.

Penjelasan untuk contoh kasus 2:

Antonio cukup memakan tiga temannya dengan skor: 3, 4, dan 5. Skor Antonio setelah ini menjadi 22. Antonio juga bisa memakan temannya yang memiliki skor: 2, 4, 5, atau 2, 3, 5. Namun tidak ada cara agar Antonio memiliki skor yang lebih besar dari Ricko dengan memakan kurang dari tiga temannya.

Penjelasan untuk contoh kasus 3:

Antonio cukup memakan dua temannya, dan hanya ada satu cara untuk melakukan ini. Pertama, ia harus memakan temannya dengan skor 50, sehingga skornya menjadi 150. Kemudian, ia harus memakan temannya yang memiliki skor 145, sehingga skornya menjadi 295.

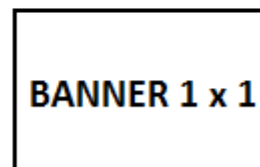
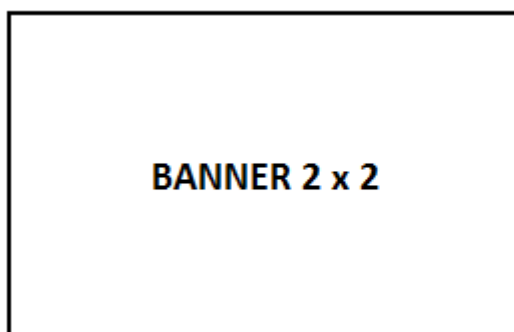
Penjelasan untuk contoh kasus 4:

Antonio tidak dapat memakan Ricko meskipun Antonio memakan seluruh teman-temannya. Lebih baik waktunya dihabiskan untuk mempersiapkan diri di UTS saja.



Problem D Steam Game

Steam adalah sebuah *portal game* yang di dalamnya terdapat ribuan permainan (*game*), dari game yang "gratisan" hingga game dengan harga yang sangat tinggi. Steam memiliki sebuah papan promosi di dalam websitenya yang berukuran lebar 4 petak dan tinggi N petak. Papan promosi ini digunakan untuk meletakkan iklan atau *banner* dari game yang dijual oleh Steam. Hanya ada dua ukuran banner yang tersedia: 1×1 petak, dan 2×2 petak.



Dengan alasan keindahan, Steam memutuskan tidak boleh ada petak yang kosong tidak terisi. Selain itu, banner game juga tidak boleh tumpang tindih dengan banner yang lainnya.

Saat ini desainer grafis dan developer dari Steam sedang mengatur bentuk papan promosi yang akan digunakan. Developer tersebut ingin meminta bantuan anda untuk menghitung ada berapa banyak kombinasi bentuk papan yang berbeda yang bisa dibentuk dari papan berukuran $4 \times N$. Perhatikan, yang diminta hanyalah banyaknya kombinasi bentuk papan yang berbeda, tidak termasuk jenis game yang ditampilkan; dengan kata lain, asumsikan semua banner diisi dengan promosi game yang sama. Lihat contoh input dan output di bawah agar lebih jelas.

Input

Input dimulai dengan sebuah bilangan bulat T ($T \leq 100.000$) yang menyatakan banyaknya kasus yang harus anda tangani. Setiap kasus berisi sebuah bilangan bulat N ($1 \leq N \leq 100.000$) dalam satu baris yang menyatakan tinggi papan promosi yang disediakan.

Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah banyak kombinasi bentuk papan promosi yang bisa dibentuk. Dikarenakan angka ini bisa sangat besar, tampilkan hanya sisa bagi (modulo) nya dengan 1.000.000.007.

Contoh input	Output untuk contoh input
2	Kasus #1: 1
1	Kasus #2: 5
2	



Penjelasan contoh kasus 1:

Hanya ada satu bentuk papan promosi yang bisa dibentuk dengan papan 4 x 1.

1 x 1	1 x 1	1 x 1	1 x 1
-------	-------	-------	-------

Penjelasan contoh kasus 2:

Ada 5 bentuk papan promosi yang bisa dibentuk dengan papan 4 x 2:

1 x 1	1 x 1	1 x 1	1 x 1	2 x 2		1 x 1	1 x 1
1 x 1	1 x 1	1 x 1	1 x 1			1 x 1	1 x 1

1 x 1	2 x 2	1 x 1	1 x 1	1 x 1	1 x 1	2 x 2
1 x 1		1 x 1		1 x 1		

2 x 2		2 x 2	



Problem E

Lukisan Pikupanda

Pikupanda adalah seekor panda yang sangat hebat dalam melukis. Cukup berikan sebuah objek, ia bisa melukis objek tersebut dengan sangat bagus hingga menyerupai foto. Kehebatannya ini tercium oleh seorang *talent scout* (pencari bakat) di Panda Academy yang kemudian memanggilnya dan memberinya sebuah tugas:

"Saya berikan kamu sebuah objek, lukislah objek tersebut semirip mungkin!"

Panda Academy akan memberikan sebuah objek yang didefinisikan dengan N buah titik. Setiap titik dinomori dari 1 sampai N , dan mempunyai koordinat kartesius (AX_i, AY_i) . Titik ke-1 (pertama) dan titik ke- N (terakhir) akan berada tepat pada sumbu X (titik dengan koordinat $y = 0$). Selain itu, titik diberikan sedemikian sehingga AX_i terurut menaik. Dengan kata lain, berlaku $AX_i < AX_{i+1}$. Lukisan dibentuk dengan menghubungkan setiap titik ke- i dengan titik ke- $i+1$ dengan garis lurus, kemudian ditutup dengan menghubungkan titik ke- N dengan titik ke-1.

Pikupanda menerima tugas Panda Academy, dan dalam sekejap berhasil melukisnya. Lukisan Pikupanda mempunyai definisi yang mirip dengan definisi objek yang diberikan oleh Panda Academy: setiap titik dinomori dari 1 sampai M dan mempunyai koordinat kartesius (BX_i, BY_i) ; titik pertama dan terakhir berada tepat pada sumbu X ; BX_i terurut menaik.

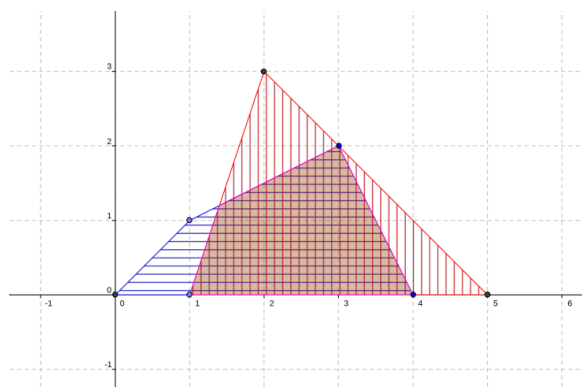
Sebelum mengumpulkan lukisannya kepada Panda Academy, Pikupanda bertemu dengan teman lamanya, Pikopanda. Ternyata Pikopanda pernah bekerja di Panda Academy, dan mengetahui metode penilaian Panda Academy. Sebagai teman lama yang baik, Pikopanda memberi tahu metode tersebut kepada Pikupanda.

Nilai kemiripan antara objek asli dan lukisan Pikupanda akan dinilai dengan rumus:

$$\text{nilai}_{\text{kemiripan}} = \frac{\text{luasWilayah}(A \cap B)}{\text{luasWilayah}(A \cup B)}$$

dimana A adalah objek asli di bidang kartesius dan B adalah lukisan yang Pikupanda buat di bidang kartesius. Luas irisan (\cap) didapat dari wilayah pada bidang kartesius yang merupakan bagian dari A dan B , sedangkan luas gabungan (\cup) didapat dari wilayah pada bidang kartesius yang merupakan bagian dari setidaknya satu di antara A dan B .

Perhatikan gambar berikut untuk lebih jelasnya:





Wilayah yang diarsir secara vertikal adalah daerah A saja, wilayah yang diarsir secara horisontal adalah daerah B saja, dan wilayah yang diarsir horisontal dan vertikal adalah irisan antara A dan B. Luas irisan antara A dan B adalah 3.8 satuan dan luas gabungan antara A dan B adalah 6.7 satuan. Sehingga Pikupanda mendapat nilai-kemiripan sebesar $3.8 / 6.7 = 0.567164$ satuan.

Pikupanda akan merasa sangat malu jika ternyata ia tidak sanggup melukis dengan cukup mirip dan ditolak oleh Panda Academy karena nilai-kemiripan yang terlalu kecil. Anda sebagai finalis BNPCHS 2015 dipercaya oleh Pikupanda untuk menghitung nilai-kemiripan lukisan Pikupanda dan objek asli sebelum ia mengumpulkan lukisannya ke Panda Academy (agar ia bisa kabur saja kalau lukisannya memang buruk). Bantulah Pikupanda!

Input

Baris pertama dari input berisi sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan dua buah bilangan bulat N dan M ($3 \leq N, M \leq 10.000$) yang secara berurutan menyatakan jumlah titik pada objek asli Panda Academy, dan jumlah titik pada lukisan Pikupanda. Empat baris berikutnya akan mendeskripsikan titik yang ada pada objek asli dan lukisan Pikupanda: baris pertama berisi N buah bilangan bulat non-negatif AX_i ; baris kedua berisi N buah bilangan bulat non-negatif AY_i ; baris ketiga berisi M buah bilangan bulat non-negatif BX_i ; baris keempat berisi M buah bilangan bulat non-negatif BY_i . Semua koordinat merupakan bilangan non-negatif, tidak melebihi 100.000, dan terurut menaik oleh koordinat X ($AX_i < AX_{i+1}$; $BX_i < BX_{i+1}$). Dijamin titik awal dan titik akhir objek maupun lukisan Pikupanda berada pada sumbu X dan luas gabungan dari objek dan lukisan lebih besar dari 0.

Output

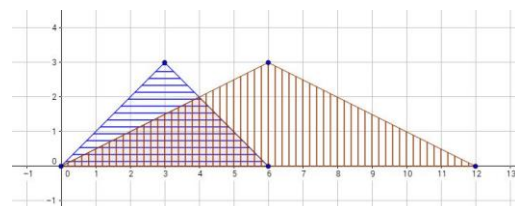
Untuk setiap kasus, output dalam satu baris "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah nilai-kemiripan dengan tingkat presisi 6 angka di belakang koma.

Contoh input	Output untuk contoh input
2 3 4 1 2 5 0 3 0 0 1 3 4 0 1 2 0 3 3 0 3 6 0 3 0 0 6 12 0 3 0	Kasus #1: 0.567164 Kasus #2: 0.285714

Penjelasan contoh kasus 1:

Kasus ini sama dengan penjelasan yang ada pada deskripsi soal di atas.

Penjelasan contoh kasus 2:



Luas A = 9
Luas B = 18
Luas $A \cap B$ = 6
Luas $A \cup B$ = 21
nilai-kemiripan = $6 / 21 = 0.285714$



Problem F

Anak Nakal

Pak Endie Vendy adalah seorang guru TK di sebuah sekolah ternama, Borobidji. Pak Endie merupakan pecinta anak-anak, sehingga ia sangat menikmati pekerjaannya. Ia menyayangi murid-muridnya bagai anaknya sendiri. Di sisi sebaliknya, anak-anak juga sangat senang diajari oleh Pak Endie karena cara mengajarnya yang unik.

Tetapi Pak Endie mempunyai cita-cita, yaitu bekerja sebagai Pissa Hatt Delivery (PHD). Demi menggapai cita-citanya, ia terpaksa memutuskan untuk meninggalkan anak-anak yang sangat dicintainya. Tidak ingin membuat anak-anak sedih, Pak Endie memutuskan untuk bermain sebuah permainan dan membagikan permen kepada anak-anak di hari terakhirnya mengajar.

Pada permainan yang dirancang Pak Endie, pertama-tama N anak dikumpulkan berdiri membentuk lingkaran. Anak-anak dinomori dari 1 sampai N , dimana anak ke- $(i+1)$ berada di sebelah kanan anak ke- i , dan anak ke-1 berada di sebelah kanan anak ke- N . Setelah itu Pak Endie membagikan kantong berisi permen ke setiap anak, dimana anak ke- i masing-masing mendapatkan kantong dengan P_i permen di dalamnya. Pak Endie sudah mempersiapkan sebuah peluit untuk permainan ini. Setiap kali Pak Endie meniupkan peluit, setiap anak harus memberikan setengah (pembulatan ke bawah) dari permen yang ada di kantongnya saat ini ke orang di sebelah kanannya. Pak Endie meniupkan peluit sejumlah K kali. Setelah K kali peniupan peluit, permainan selesai dan masing-masing anak membawa pulang semua permen yang ada di kantongnya.

Sebagai contoh misalkan terdapat 3 anak dengan masing-masing kantong berisi 3, 2, 5 permen. Ketika peluit ditiup:

- Anak ke-1 akan memberikan 1 permen ($3/2$ dibulatkan ke bawah = 1) kepada anak ke-2
- Anak ke-2 akan memberikan 1 permen ($2/2 = 1$) kepada anak ke-3
- Anak ke-3 akan memberikan 2 permen ($5/2$ dibulatkan ke bawah = 2) kepada anak ke-1

Sehingga setelah peluit ditiup masing-masing anak akan mempunyai kantong dengan 4, 2, 4 permen.

Di luar dugaan Pak Endie, terdapat 3 anak nakal yang tidak mengikuti peraturan: Andi, Budi, dan Cedi.

- Andi melakukan pembulatan ke atas ketika memberikan permen ke anak sebelahnyanya. Contoh: jika jumlah permen di kantong Andi sekarang adalah 5, dia akan memberikan 3 permen (seharusnya 2).
- Jika kantong permen Budi tidak kosong, Budi akan memakan 1 buah permen terlebih dahulu sebelum memberikan permen ke anak sebelahnyanya. Contoh: jika jumlah permen di kantong Budi sekarang adalah 6, dia akan memakan 1 permen (menjadi 5), kemudian memberikan 2 permen ke anak di sebelahnyanya.
- Cedi tetap mengikuti aturan pembulatan ke bawah, namun dia hanya memberikan maksimum 10 permen. Dengan kata lain, jika permen yang harus diberikan menurut aturan lebih besar dari 10, dia hanya akan memberikan 10 permen. Jika permen yang harus diberikan menurut aturan tidak lebih besar dari 10, maka tidak ada perubahan pada permen yang diberikan Cedi. Contoh: jika jumlah permen di kantong Cedi sekarang adalah 23, dia hanya akan memberikan 10 (seharusnya 11). Contoh lain jika jumlah permen di kantong Cedi sekarang adalah 12, Cedi tetap memberikan 6 permen sesuai aturan.



Pak Endie tidak memberikan hukuman apa-apa mengenai kelakuan nakal ketiga anak tersebut. Tetapi sekarang Pak Endie ingin mengetahui: berapa permen yang dibawa pulang masing-masing anak nakal tersebut?

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan lima buah bilangan bulat N, A, B, C, K ($3 \leq N \leq 100.000$; $1 \leq A, B, C \leq N$; $A \neq B \neq C$; $0 \leq K \leq 1.000$) yang secara berurutan menyatakan banyaknya anak yang diajari Endie, nomor masing-masing anak nakal Andi, Budi, Cedi, dan jumlah tiupan peluit. Baris berikutnya berisi N buah bilangan P_i ($0 \leq P_i \leq 1.000$) secara berurutan, yaitu jumlah permen pada kantong yang dibagikan Pak Endie kepada anak ke- i pada awal permainan.

Output

Untuk setiap kasus, output dalam satu baris "Kasus #X: PA PB PC" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan PA PB PC adalah permen yang didapatkan masing-masing oleh Andi, Budi, dan Cedi.

Contoh input	Output untuk contoh input
4 4 3 2 4 1 4 2 3 25 4 3 2 4 2 4 2 3 25 5 1 3 5 4 5 2 1 5 7 3 1 2 3 3 1 2 3	Kasus #1: 1 3 17 Kasus #2: 1 7 10 Kasus #3: 3 4 2 Kasus #4: 0 2 1

Penjelasan contoh kasus 1:

Terdapat 4 anak, dan Pak Endie akan meniup peluit sebanyak 1 kali. Andi, Budi, dan Cedi masing-masing adalah anak ke-3, ke-2, dan ke-4. Mula-mula jumlah permen di kantong setiap anak secara berurutan adalah {4, 2, 3, 25}. Ketika peluit ditiup:

- Anak ke-1 memberikan 2 permen ($4/2 = 2$) kepada anak ke-2 (Budi).
- Anak ke-2 (Budi) memakan 1 permen terlebih dahulu, sehingga tersisa 1 permen. Kemudian memberikan 0 permen ($1/2$ dibulatkan ke bawah = 0) kepada anak ke-3 (Andi).
- Anak ke-3 (Andi) memberikan 2 permen ($3/2$ dibulatkan ke atas) kepada anak ke-4.
- Anak ke-4 (Cedi) seharusnya memberikan 12 permen ($25/2$ dibulatkan ke bawah). Tetapi karena anak ini merupakan Cedi, yang diberikan hanya 10 permen.

Jumlah permen di kantong setiap anak setelah permainan berakhir adalah {12, 3, 1, 17}. Maka permen yang dimiliki Andi, Budi, dan Cedi masing-masing adalah 1 (anak ke-3), 3 (anak ke-2), dan 17 (anak ke-4).

Penjelasan contoh kasus 3:

Mula-mula jumlah permen di kantong setiap anak secara berurutan adalah {5, 2, 1, 5, 7}. Berikut yang terjadi pada kantong permen setiap anak setelah setiap tiupan peluit:

- Setelah tiupan peluit ke-1: {5, 4, 1, 3, 6}
- Setelah tiupan peluit ke-2: {5, 5, 2, 2, 4}
- Setelah tiupan peluit ke-3: {4, 6, 3, 1, 3}
- Setelah tiupan peluit ke-4: {3, 5, 4, 2, 2}

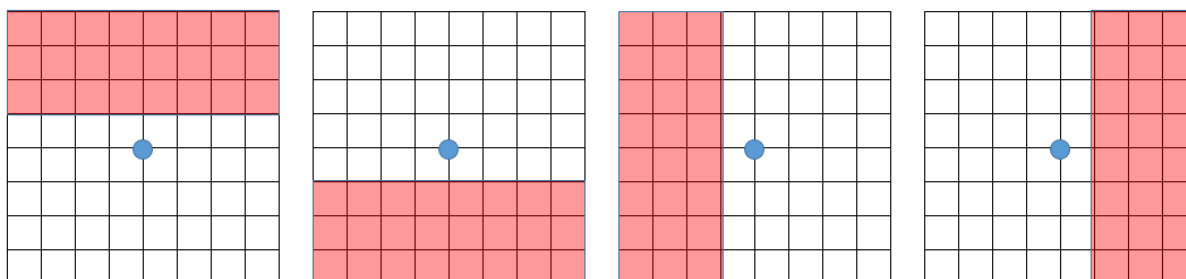


Problem G

Titik Aman

Vincentius mendapat sebuah *video game* terbaru dari orang tuanya yang berjudul *Pointless*. Game ini mengenai dunia titik 2 dimensi yang luasnya tak terhingga, dan di dalamnya terdapat N titik jahat yang siap menembakkan sinar laser. Setiap titik jahat memiliki 4 opsi penembakan laser. Asumsikan ada sebuah titik jahat pada koordinat (X, Y) , maka 4 opsi tersebut adalah:

- (U) UP SHOOT : Semua titik (X', Y') dengan $Y' > Y$ akan mati.
- (D) DOWN SHOOT : Semua titik (X', Y') dengan $Y' < Y$ akan mati.
- (L) LEFT SHOOT : Semua titik (X', Y') dengan $X' < X$ akan mati.
- (R) RIGHT SHOOT : Semua titik (X', Y') dengan $X' > X$ akan mati.



UP SHOOT

DOWN SHOOT

LEFT SHOOT

RIGHT SHOOT

Selain N buah titik jahat tersebut, pemain juga diberikan sebuah koordinat titik aman. Titik aman ini tidak boleh terkena laser (terserang) oleh titik jahat manapun. Dalam game ini, Vincentius harus menentukan opsi dari masing-masing titik jahat sehingga tidak ada satupun yang menyerang titik aman.

Vincentius akan memainkan game ini sebanyak Q kali, dan pada masing-masing game, Vincentius diberikan satu koordinat titik aman. Bantulah Vincentius untuk menghitung berapa banyak konfigurasi opsi titik jahat agar tidak ada satupun laser dari titik jahat yang mengenai titik aman. Dua konfigurasi dikatakan berbeda jika dan hanya jika terdapat setidaknya satu titik jahat dengan opsi yang berbeda. Perlu diperhatikan, pada masing-masing game:

- Jumlah dan lokasi masing-masing titik jahat tidak berubah.
- Titik aman akan hilang pada akhir game dan digantikan dengan titik aman yang baru pada game selanjutnya; dengan kata lain, hanya terdapat satu titik aman pada setiap game.

Anda boleh mengasumsikan bahwa tidak ada dua titik jahat dengan koordinat yang sama, dan titik aman tidak akan terletak pada koordinat yang sama dengan titik jahat manapun. Sinar laser dari titik jahat tidak bisa menghancurkan titik jahat lainnya (titik jahat kebal terhadap serangan laser).

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan sebuah bilangan bulat N ($0 \leq N \leq 20.000$) yang menyatakan banyaknya titik jahat. N baris berikutnya masing-masing terdiri dari 2 buah bilangan bulat X_i dan Y_i ($1 \leq X_i, Y_i \leq 100.000$) yang menyatakan koordinat X dan Y titik jahat ke- i secara berurutan. Baris berikutnya terdiri dari sebuah bilangan bulat Q ($1 \leq Q \leq 20.000$) yang menyatakan banyaknya game yang dimainkan Vincentius. Q baris berikutnya masing-masing terdiri dari 2 buah



bilangan bulat XA_i dan YA_i ($1 \leq XA_i, YA_i \leq 100.000$) yang menyatakan koordinat X dan Y titik aman ke-i secara berurutan.

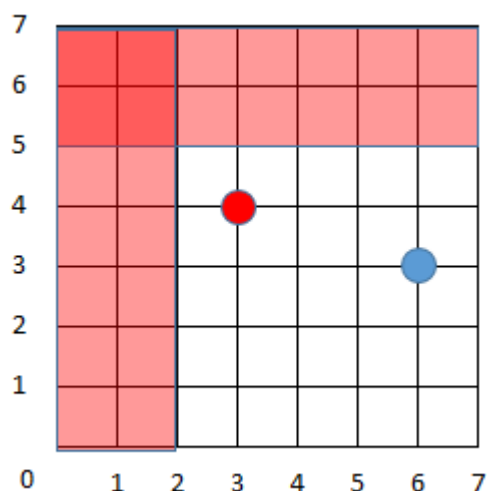
Output

Untuk setiap kasus, cetak dalam satu baris "Kasus #X:" (tanpa tanda kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan. Q baris berikutnya masing-masing berisi sebuah bilangan bulat yang menyatakan banyaknya konfigurasi opsi titik jahat pada masing-masing game. Karena jawabannya bisa sangat besar, modulo hasilnya dengan 1.000.000.007.

Contoh input	Output untuk contoh input
2 1 3 4 2 6 3 1 1 2 5 5 3 4 2 4 3 5 2	Kasus #1: 2 2 Kasus #2: 4 6

Penjelasan contoh kasus 1:

Kondisi pada game pertama adalah sebagai berikut:



Hanya ada dua opsi yang mungkin bagi titik jahat yang berada pada (3, 4), yaitu U (UP SHOOT) dan L (LEFT SHOOT). Konfigurasi opsi lainnya akan mengenai titik aman.

Begitu juga dengan game kedua, hanya ada 2 konfigurasi opsi yang bisa digunakan.

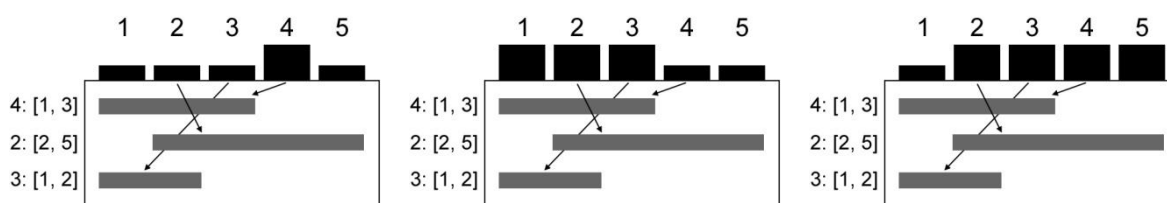
Penjelasan contoh kasus 2:

Pada game pertama, 4 konfigurasi opsi untuk titik jahat (5, 5) dan (3, 4) adalah: UU, UL, RU, dan RL. Pada game kedua, 6 konfigurasi opsi yang bisa digunakan adalah: UU, UL, LU, LL, RU, dan RL.



Problem H Permainan Pandy

Pandy adalah seekor panda yang gemar bermain dan membuat permainan yang aneh (tapi menarik). Pandy memiliki N buah tombol yang dibariskan dalam satu barisan dan diberi nomor dari 1 hingga N secara berurutan. Kondisi awal semua tombol yang ada adalah turun (tertekan), kecuali tepat 1 buah tombol, yaitu tombol S . Setiap tombol (A) terhubung dengan maksimal 1 buah panel dengan rentang $[B, C]$. Apabila tombol A ditekan, maka tombol-tombol dari B hingga C akan naik, dan semua tombol lain yang ada akan turun. Tombol yang dapat ditekan tentunya hanyalah tombol yang sedang dalam keadaan naik, dan sistem ini memiliki mekanisme yang mencegah pemain untuk menekan tombol yang sama lebih dari satu kali secara berurutan.



Pada gambar di atas (kiri), ada 5 tombol (dari kiri ke kanan: 1..5) dengan awalnya tombol yang naik adalah tombol 4. Tombol 4 terhubung dengan panel $[1, 3]$, tombol 2 terhubung dengan panel $[2, 5]$, dan tombol 3 terhubung dengan panel $[1, 2]$; tombol 1 dan 5 tidak terhubung dengan panel apapun. Pada dua gambar berikutnya, kita bisa melihat dua kali penekanan secara berturut-turut:

1. Tekan tombol 4 \rightarrow tombol 1, 2, dan 3 menjadi naik; tombol 4 dan 5 turun.
2. Tekan tombol 2 \rightarrow tombol 2, 3, 4, dan 5 menjadi naik; tombol 1 turun.

Tombol yang naik setelah dua kali penekanan tersebut adalah tombol 2, 3, 4, dan 5.

Setiap kali pemain menekan tombol (A), maka pemain akan mendapatkan poin sebesar P_A . Anda diberikan N buah tombol dengan rangkaian panelnya, dan tiga buah bilangan bulat E , P_{\min} , dan P_{\max} , serta Q buah query berupa bilangan bulat S . Untuk setiap query S , anda diminta untuk mencari urutan penekanan tombol di mana:

- S adalah satu-satunya tombol yang naik pada awal permainan.
- Tombol E harus naik pada akhir permainan.
- Total poin yang dikumpulkan pemain berada pada rentang P_{\min} dan P_{\max} .

Jika ada lebih dari satu urutan penekanan yang demikian, output urutan yang paling kecil secara leksikografis (urutan kamus).

Seperti yang kita ketahui, panda adalah binatang yang dilindungi. Pandy tidak mau makan sebelum pertanyaan (query) yang diberikan berhasil dijawab dengan benar. Kelangsungan hidup Pandy berada di tangan anda.

Input

Baris pertama berisi sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya kasus yang harus anda tangani. Input untuk setiap kasus:

- Baris pertama terdiri dari 3 buah bilangan bulat N , M dan Q ($1 \leq M \leq N \leq 1.000$; $1 \leq Q \leq 50$) yang menyatakan banyaknya tombol, banyaknya panel, dan banyaknya query secara berurutan.
- Baris kedua terdiri dari N buah bilangan bulat P_i ($1 \leq P_i \leq 1.000$) yang menyatakan poin yang akan didapatkan pemain setiap kali tombol ke i ditekan.



- Baris ketiga terdiri dari 3 buah bilangan bulat E, P_{\min}, P_{\max} ($1 \leq E \leq N$; $1 \leq P_{\min} \leq P_{\max} \leq 1.000$) yang menyatakan tombol yang harus naik pada akhir permainan, total poin minimum, dan total poin maksimum secara berurutan.
- M baris berikutnya masing-masing terdiri dari tiga buah bilangan bulat A_i, B_i, C_i ($1 \leq A_i, B_i, C_i \leq N$; $B_i \leq C_i$), yang merepresentasikan panel $[B_i, C_i]$ yang terhubung dengan tombol A_i .
- Q baris berikutnya masing-masing terdiri dari sebuah bilangan bulat S ($1 \leq S \leq N$) yang menyatakan tombol yang mula-mula dalam keadaan naik pada setiap query.

Output

Baris pertama dari setiap kasus adalah "Kasus #X:" (tanpa kutip), dimana X adalah nomor kasus, dimulai dari 1. Q baris berikutnya masing-masing berisi jawaban dari setiap query yang diberikan pada kasus yang bersangkutan secara berurutan. Setiap baris berisi urutan penekanan tombol yang menjawab query yang diberikan; masing-masing bilangan dipisahkan oleh sebuah spasi. Apabila tidak ada urutan penekanan yang dapat menjawab query yang diberikan, output -1 dalam satu baris untuk query tersebut.

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

Pada penjelasan di bawah, notasi $\{a, b, c, \dots\}$ artinya tombol yang berada dalam keadaan naik pada saat itu adalah tombol a , tombol b , tombol c , dan seterusnya.

Penjelasan untuk contoh kasus 1:

Kasus satu terdiri dari 3 query dengan $E = 3$, $P_{\min} = 5$, dan $P_{\max} = 7$.

- Query #1: $S = 5$. Tombol 5 tidak terhubung dengan panel apapun, sehingga tidak ada cara agar tombol 3 bisa dalam keadaan naik.
- Query #2: $S = 3$. $\{3\} \rightarrow$ tekan 3 $\rightarrow \{1, 2\} \rightarrow$ tekan 2 $\rightarrow \{2, 3, 4\}$. Total poin yang dikumpulkan adalah: 3 (dari penekanan pertama) + 2 (dari penekanan kedua) = 5.
- Query #3: $S = 2$. $\{2\} \rightarrow$ tekan 2 $\rightarrow \{2, 3, 4\} \rightarrow$ tekan 3 $\rightarrow \{1, 2\} \rightarrow$ tekan 2 $\rightarrow \{2, 3, 4\}$. Total poin yang dikumpulkan adalah: $2 + 3 + 2 = 7$. Urutan penekanan "2 4" juga memenuhi syarat yang diberikan, namun "2 3 2" lebih kecil secara leksikografis.

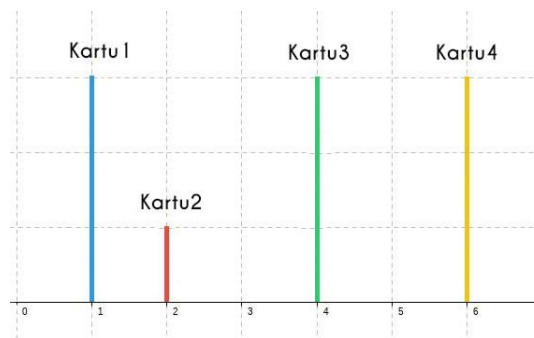


Problem I Domino Rex

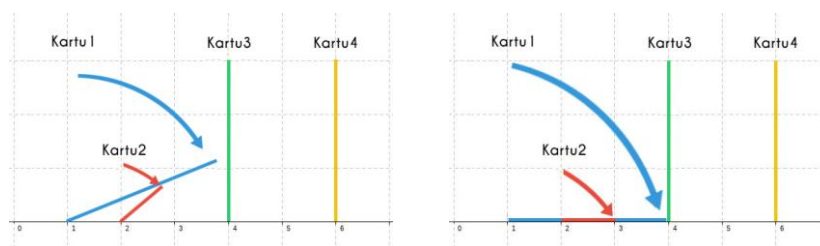
Monde Bë baru saja pulang dari liburannya di pulau Cissa Rùj. Liburannya sangat menyenangkan, ia melihat banyak binatang-binatang unik dan langka yang hidup hanya di pulau tersebut. Banyak cendera mata yang dijual disana. Monde Bë memutuskan untuk membeli sebuah cendera mata yang dinilainya paling unik, yaitu *Domino Rex*.

Domino Rex adalah sejenis permainan strategi menjatuhkan kartu. Mula-mula, N buah kartu didirikan secara tegak di posisi-posisi tertentu pada sebuah papan (garis lurus). Kartu tersebut memiliki nomor dari 1 hingga N , di mana kartu ke- i memiliki tinggi H_i cm dan diletakkan pada posisi P_i cm dari ujung kiri papan.

Berikut contoh ilustrasi untuk $N = 4$, $H = \{3, 1, 3, 3\}$, dan $P = \{1, 2, 4, 6\}$.

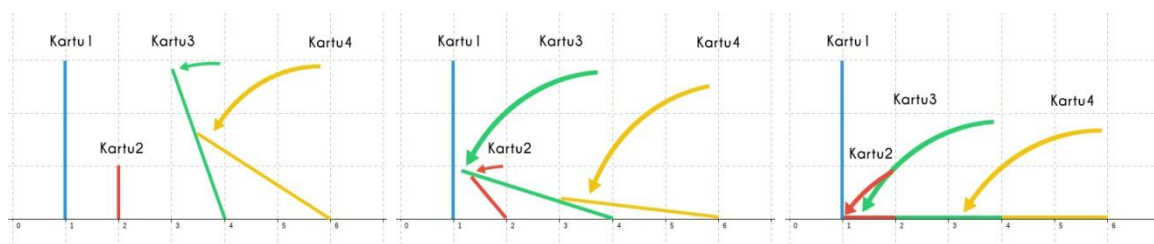


Pemain tidak bisa menggeser posisi kartu yang diberikan, namun ia bisa menjatuhkan kartu-kartu tersebut, baik ke arah kiri maupun ke arah kanan. Kartu yang jatuh (A) akan menjatuhkan kartu yang lain (B) jika arah jatuh kartu A adalah menuju kartu B dan tinggi kartu A **lebih besar** dari jarak A menuju B; arah jatuh kartu B (jika bisa dijatuhkan) akan sama dengan arah jatuh kartu A. Ilustrasi berikut menggambarkan proses jatuhnya kartu 1 ke kanan.



Ketika kartu 1 dijatuhkan, kartu 2 juga ikut terjatuh; tetapi kartu 3 tidak terjatuh karena, di antara kartu 1 maupun kartu 2 (kartu-kartu yang terjatuh), tidak ada yang bisa menjatuhkan kartu 3; tinggi kartu 1 tepat sama dengan jaraknya menuju kartu 3, sedangkan tinggi kartu 2 lebih kecil dari jaraknya ke kartu 3.

Perhatikan juga bahwa proses jatuhnya kartu memiliki sifat berantai; dengan kata lain, kartu yang terjatuh (tidak harus dijatuhkan langsung) juga bisa menjatuhkan kartu lainnya selama syarat jatuhnya kartu terpenuhi. Pada contoh kasus di atas, jika kita menjatuhkan kartu 4 ke arah kiri, maka kartu 3 dan kartu 2 juga akan terjatuh; kartu 3 dijatuhkan oleh kartu 4, sedangkan kartu 2 dijatuhkan oleh kartu 3.



Di dalam permainan Domino Rex, pemain harus membuat semua kartu terjatuh. Semakin sedikit kartu yang dijatuhkan secara langsung oleh pemain, maka nilai yang diperoleh juga semakin besar. Monde Bë ingin mendapatkan nilai sebesar mungkin dalam permainan ini. Bantu Monde Bë untuk menghitung berapa jumlah kartu minimal yang perlu dijatuhkan oleh Monde Bë secara langsung agar semua kartu yang ada terjatuh.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 200$) yang menyatakan banyaknya kasus yang harus anda tangani. Baris pertama dari setiap kasus merupakan sebuah bilangan bulat N ($1 \leq N \leq 12.000$). N baris berikutnya masing-masing baris berisi dua bilangan bulat P_i dan H_i ($0 \leq P_i \leq 100.000$; $1 \leq H_i \leq 100.000$), yaitu posisi dan tinggi kartu ke- i secara berurutan. Dijamin tidak ada dua kartu yang berada di posisi yang sama.

Output

Untuk setiap kasus, cetak dalam satu baris, "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah jumlah kartu minimal yang perlu dijatuhkan secara langsung oleh Monde Bë.

Contoh input	Output untuk contoh input
3 4 1 3 2 1 4 3 6 3 3 1 2 2 2 3 1 3 1 1 2 1 3 1	Kasus #1: 2 Kasus #2: 1 Kasus #3: 3

Penjelasan untuk contoh input 1:

Diperlukan setidaknya dua kartu untuk dijatuhkan secara langsung: kartu 1 dan kartu 4.

Penjelasan untuk contoh input 2:

Cukup jatuhkan kartu 1 ke arah kanan; efek berantainya akan menjatuhkan kartu 2 dan kartu 3.



Problem J Tree Mall

Sebuah mall baru yang unik dan bernama Tree Mall baru saja dibuka di Tangerang. Mall ini memiliki N buah toko yang dinomori dari 1 hingga N . Berbeda dengan mall pada umumnya, toko-toko di dalam mall ini dihubungkan oleh tepat $N - 1$ buah jalan/lorong di mana setiap jalan menghubungkan tepat hanya dua buah toko yang berbeda. Untuk berpergian dari satu toko ke toko lainnya, pengunjung harus menggunakan jalan yang tersedia, dan mungkin saja pengunjung harus mengunjungi beberapa toko terlebih dahulu untuk sampai di toko yang ingin dikunjungi. Namun dipastikan selalu ada jalan (baik langsung maupun tidak langsung) dari satu toko ke semua toko yang ada. Hal unik lainnya dari mall ini adalah: pintu masuk dan keluar mall ini hanya terdapat pada setiap toko yang hanya memiliki tepat satu buah jalan yang terhubung ke sebuah toko lainnya.

Seperti biasa, semakin kita menjauhi pintu masuk mall, maka sinyal telepon genggam yang diterima akan semakin berkurang (toko-toko pada pintu masuk/keluar mall memiliki sinyal penuh). Vincent yang secara kebetulan sedang berada di Tree Mall ingin menghubungi temannya dengan telepon genggam untuk menemaninya belanja peralatan sepak bola; namun Vincent mengalami kesulitan karena pada beberapa toko tertentu, sinyal yang didapat telepon genggamnya sangat sedikit. Dari hasil pengamatan Vincent, sinyal telepon genggam akan berkurang 1 setiap kali ia menjauh satu jalan dari pintu masuk/keluar terdekat. Dengan kata lain, banyaknya sinyal yang hilang pada sebuah toko adalah banyaknya jalan minimum yang perlu digunakan untuk mencapai pintu masuk/keluar mall tersebut.

Vincent ingin segera menghubungi temannya dan memberitahukan posisinya. Namun sebelumnya, Vincent perlu mengetahui berapa banyak sinyal yang hilang pada masing-masing toko di dalam Tree Mall.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan sebuah bilangan bulat N ($1 \leq N \leq 20.000$) yang menyatakan banyaknya toko pada Tree Mall. $N-1$ baris yang masing-masing berisi A dan B ($1 \leq A, B \leq N$; $A \neq B$) yang menyatakan ada jalan yang menghubungkan toko A dan toko B .

Output

Untuk setiap kasus, output dalam satu baris "Kasus #X:" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan. Baris kedua dari setiap output berisi N buah bilangan bulat yang menyatakan banyaknya sinyal yang berkurang pada toko ke- i secara berurutan. Masing-masing bilangan bulat tersebut dipisahkan oleh sebuah spasi.

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



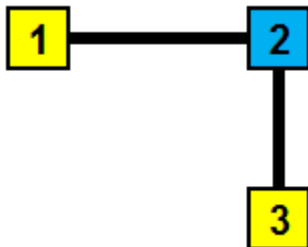
5	
1 2	
2 3	
2 4	
4 5	

Penjelasan contoh kasus 1:



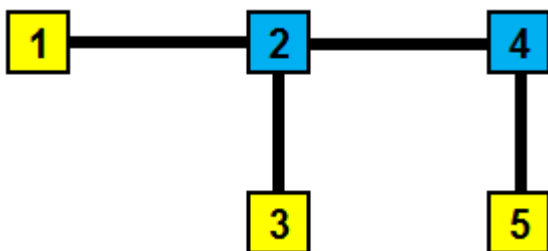
Setiap toko merupakan pintu masuk/keluar Tree Mall, sehingga masing-masing toko tidak mengalami pengurangan sinyal.

Penjelasan contoh kasus 2:



Pada kasus ini, pintu masuk/keluar Tree Mall ada pada toko 1 dan 3. Toko 2 mengalami pengurangan sinyal sebesar 1 karena toko 2 memerlukan 1 jalan untuk mencapai pintu masuk/keluar terdekat.

Penjelasan contoh kasus 3:



Pada kasus ini, pintu masuk/keluar Tree Mall berada pada toko 1, 3, dan 5. Jika dilihat dari sinyal yang masuk dari pintu pada toko 5, maka toko 2 mengalami pengurangan 2 sinyal; namun jika dilihat dari pintu pada toko 1 atau 3, maka toko 2 hanya mengalami pengurangan 1 sinyal. Oleh karena itu, toko 2 hanya mengalami pengurangan 1 sinyal, karena sinyal yang masuk dari pintu pada toko 1 dan 3 lebih kuat. Begitu juga dengan toko nomor 4. Toko 4 mengalami pengurangan 1 sinyal dari pintu terdekat, yaitu pintu pada toko 5.



Problem K

Membagi Uang

Kota Zakrata cukup terkenal dengan kelompok mafianya. Setiap hari ada ratusan keributan yang ditimbulkan oleh organisasi-organisasi mafia ini; tidak jarang keributan yang ditimbulkan disebabkan karena perselisihan antar organisasi yang ada dalam menjalankan “bisnis” mereka. Aliansi mafia dibentuk untuk menyelesaikan permasalahan antar mafia yang kerap terjadi. Tentunya semakin sering perselisihan timbul, “keuntungan” yang bisa diraup oleh para mafia semakin berkurang.

Alih-alih menyelesaikan perselisihan, aliansi mafia mengusulkan sebuah metode unik untuk mengatur bisnis bagi para organisasi mafia. Organisasi-organisasi mafia tidak keberatan, karena tujuan mereka adalah uang. Pertama-tama, aliansi akan menyiapkan dana sebesar N untuk setiap organisasi mafia yang ingin mengikuti metode ini. Organisasi mafia yang mengikuti harus mengirimkan sebanyak P pekerja. Dana dari aliansi akan diberikan secara sama rata kepada P ($P < N$) pekerja, dan memungkinkan ada sisa dana (sebesar S , di mana $0 \leq S < P$). Kemudian para pekerja tersebut dibagi menjadi K ($1 < K$) buah tim, di mana setiap tim harus memiliki jumlah anggota yang sama banyak. Sisa dana (S) tersebut kemudian dibagikan ke K tim secara merata, dan tidak boleh ada dana yang tersisa. Ini semua adalah syarat dari aliansi yang harus dipenuhi oleh organisasi mafia yang ingin mengikuti metode ini.

Tiap tim yang dibentuk akan berlomba mengembangkan dana yang diberikan sehingga memberikan keuntungan yang sebesar-besarnya dalam waktu enam bulan; tim yang memiliki keuntungan terbesar yang menang. Tergantung kebijakan internal di setiap organisasi mafia, anggota dari tim ini bisa saja dinaikan pangkatnya.

Beberapa organisasi mafia tertarik dengan metode yang diusulkan oleh aliansi mafia. Namun, mereka kesulitan menentukan berapa pekerja yang harus mereka kirimkan agar syarat dari aliansi mafia bisa dipenuhi. Tentunya mereka juga ingin mengirimkan sebanyak mungkin pekerja.

Anda didekati oleh aliansi mafia untuk membantu organisasi mafia yang ingin mengikuti metode ini. Tentukan banyak pekerja yang diperlukan agar syarat dari aliansi mafia bisa dipenuhi. Asumsikan organisasi-organisasi mafia memiliki anggota yang luar biasa banyaknya (semua bisa jadi pekerja).

Input

Baris pertama berisi sebuah bilangan bulat T ($1 \leq T \leq 100.000$) yang menyatakan banyaknya kasus. Setiap kasus berisi sebuah bilangan bulat N ($1 \leq N \leq 1.000.000$) yang merepresentasikan banyaknya dana yang disiapkan oleh aliansi mafia.

Output

Untuk setiap kasus, output dalam satu baris “Kasus #X: Y” (tanpa kutip), di mana X adalah nomor kasus dimulai dari 1, dan Y adalah jumlah pekerja maksimal yang diperlukan agar syarat dari aliansi mafia bisa dipenuhi. Jika tidak ada nilai P (jumlah pekerja) di mana syarat aliansi mafia bisa dipenuhi, output -1 untuk nilai Y .



BNPC

The 2015
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
3 15 2 1001	Kasus #1: 12 Kasus #2: -1 Kasus #3: 994

Penjelasan contoh kasus 1:

Dengan 12 pekerja, setiap pekerja masing-masing akan mendapatkan dana sebesar 1, dan tersisa dana sebesar 3. Kemudian pekerja tersebut dibagi ke dalam 3 tim yang masing-masing mendapatkan dana sebesar 1 dari sisa dana yang sebelumnya.

Penjelasan contoh kasus 2:

Syarat dari aliansi mafia tidak akan bisa dipenuhi jika dana yang disiapkan adalah 2. Jika organisasi mafia mengirimkan 1 pekerja, maka pekerja tersebut akan mendapatkan semua dana yang ada dan tidak ada dana tersisa. Jika pekerja yang dikirimkan adalah 2, masing-masing akan mendapatkan dana sebesar 1, dan tidak ada dana sisa.