



Problem A

Duel Maut

Sebuah pulau kecil di Indonesia baru saja ditemukan oleh dua kelompok bajak laut. Ternyata pulau ini dahulu adalah tempat habitat rahasia panda yang kini menyimpan segunung harta karun, yang jika dirupiahkan, mungkin sekitar satu triliun rupiah. Dua kelompok bajak laut ini langsung bersitegang begitu mereka menemukan harta karun ini. Mereka adalah bajak laut yang serakah; mereka tidak rela berbagi harta ini dengan kelompok lain. Kedua kapten dari masing-masing kelompok bajak laut sepakat untuk menyelesaikan sengketa ini melalui duel maut, sebuah tradisi kuno para bajak laut.

Duel maut adalah duel tangan kosong (tanpa senjata api atau senjata tajam) yang dilakukan oleh dua petarung dari kelompok bajak laut yang berbeda. Masing-masing kelompok akan mengirimkan satu perwakilan untuk mengikuti duel maut ini. Pemenang dari duel maut ini adalah pemegang keputusan akhir dari apapun masalah yang mereka peributkan. Keputusan ini harus dipatuhi, atau mereka harus berhadapan dengan raja bajak laut, yang tentu saja resikonya lebih menakutkan.

Kekuatan dari masing-masing petarung bajak laut bisa direpresentasikan dengan sebuah bilangan bulat, di mana semakin besar bilangannya, maka semakin kuat petarung tersebut. Duel maut ini tentunya semakin seru jika dua petarung yang mengikuti duel memiliki kekuatan yang (hampir) berimbang. Diberikan data kekuatan masing-masing petarung dari dua kelompok bajak laut, tentukan selisih terkecil dari kekuatan dua petarung (satu dari setiap kelompok) yang bisa ditemukan.

Contoh.

Kelompok bajak laut A memiliki 5 petarung: 10, 9, 41, 17, 24.

Kelompok bajak laut B memiliki 4 petarung: 50, 19, 29, 51.

		A				
		10	9	41	17	24
B	50	40	41	9	33	26
	19	9	10	22	2	5
	29	19	20	12	12	5
	51	41	42	10	34	27

Tabel 1. Selisih kekuatan

Tabel 1 menunjukkan selisih kekuatan masing-masing petarung di kelompok bajak laut A dengan petarung di kelompok bajak laut B. Dari tabel tersebut, kita bisa melihat bahwa selisih terkecil adalah 2, jika petarung dengan kekuatan 17 dari kelompok A bertarung dengan petarung dengan kekuatan 19 dari kelompok B.

Input

Baris pertama dari input adalah 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 ($1 \leq N, M \leq 100$) yang menyatakan banyaknya petarung di kedua kelompok bajak laut secara berurutan. Baris berikutnya terdiri dari N bilangan bulat A_i ($1 \leq A_i \leq 1.000$) yang menyatakan kekuatan petarung di kelompok bajak laut pertama. Baris berikutnya terdiri dari M bilangan bulat B_i ($1 \leq B_i \leq 1.000$) yang menyatakan kekuatan petarung di kelompok bajak laut kedua.



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 selisih terkecil dari kekuatan dua petarung yang bisa ditemukan dari masing-masing kelompok bajak laut pada kasus tersebut.

Contoh input	Output untuk contoh input
4 5 4 10 9 41 17 24 50 19 29 51 1 1 10 100 3 1 100 100 100 100 5 5 100 200 300 400 500 105 203 205 270 299	Kasus #1: 2 Kasus #2: 90 Kasus #3: 0 Kasus #4: 1

Penjelasan contoh kasus 2

Hanya ada satu pasang petarung yang bisa mengikuti duel maut ini, selisih kekuatan mereka adalah $100 - 10 = 90$.

Penjelasan contoh kasus 3

Semua petarung memiliki kekuatan yang sama, sehingga selisih terkecil kekuatan yang bisa ditemukan adalah 0.

Penjelasan contoh kasus 4

Selisih terkecil didapatkan dari pasangan petarung dengan kekuatan 300 dari kelompok bajak laut pertama dan petarung dengan kekuatan 299 dari kelompok bajak laut kedua.



Problem B**Belanja Baju Bambu Baru**

Panci, seekor panda perempuan, ingin sekali membeli baju bambu baru untuk menyambut hari Panda Nasional tahun ini (sama halnya dengan kita yang ingin memiliki baju baru di setiap perayaan tahun baru). Ia pun mengajak Randi, kekasih pandanya, untuk mencari baju baru tersebut di Mal Super Panda. Panci memiliki kebiasaan yang unik: ia selalu membeli baju yang paling murah yang ia temui di antara semua toko yang ia kunjungi, dan nilai kesenangan Panci adalah selisih harga baju yang ia beli dengan harga rata-rata baju dari semua toko yang ia kunjungi (semakin besar selisihnya, maka ia semakin senang); dan Randi mengetahui kebiasaan unik Panci yang satu ini.

Mal Super Panda memiliki susunan toko-toko yang cukup unik:

1. Mal Super Panda terdiri dari L lantai, dari lantai 1 hingga lantai L .
2. Setiap lantai K memiliki tepat 2^{K-1} toko untuk semua K ($1 \leq K \leq L$), sehingga total terdapat $2^L - 1$ toko di Mal Super Panda.
3. Setiap toko memiliki nomor berurutan dari 1 yang dimulai dari lantai bawah ke lantai atas dan dari kiri ke kanan secara berurutan.
4. Setiap toko bernomor P hanya terhubung (oleh eskalator) dengan maksimal 3 toko lain:
 - a. Toko bernomor $P / 2$ (bulatkan ke bawah) yang ada di lantai di bawahnya ¹,
 - b. Toko bernomor $2 * P$ yang ada di lantai di atasnya ²,
 - c. Toko bernomor $2 * P + 1$ yang ada di lantai di atasnya ².untuk semua P . Tidak ada jalan lain untuk mengunjungi toko-toko yang ada selain menggunakan eskalator yang baru saja dijelaskan.
¹ kecuali ketika $P = 1$ di mana tidak ada toko lain di lantai di bawahnya.
² kecuali ketika toko bernomor P berada di lantai L (paling atas).
5. Harga baju di toko P selalu lebih mahal dibandingkan harga baju di toko $2 * P$ dan toko $2 * P + 1$.

Randi dan Panci bisa memulai pencarian baju dari toko manapun, tidak harus dari pintu masuk di toko 1 (yang terjadi sebenarnya: Randi akan menutup mata Panci begitu memasuki Mal Super Panda dan membukanya ketika mereka tiba di toko yang Randi mau). Mereka bisa berjalan-jalan di dalam Mal tersebut dan mengunjungi toko-toko yang ada selama toko yang dikunjungi terhubung; Randi tidak akan menutup mata Panci lagi begitu mereka mulai berjalan-jalan. Panci akan memeriksa harga baju di setiap toko yang ia kunjungi. Sebuah toko boleh saja dikunjungi lebih dari satu kali, namun harga baju di toko tersebut hanya dihitung satu kali.

Panci mempercayakan rute perjalanan mereka kepada Randi. Bantu Randi untuk menentukan rute yang bisa memberikan nilai kesenangan terbesar bagi Panci (output nilai kesenangannya saja).

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 10$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan sebuah bilangan bulat L ($1 \leq L \leq 15$) dalam satu baris yang menyatakan banyaknya lantai pada Mal Super Panda. Baris berikutnya terdiri dari $2^L - 1$ bilangan bulat H_i ($1 \leq H_i \leq 1.000.000$) yang menyatakan harga baju pada toko ke- i secara berurutan dengan $i = 1 \dots 2^L - 1$. Harga baju $H_{2^i} < H_i$ dan $H_{2^i+1} < H_i$ untuk semua i (lihat penjelasan soal di atas).



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 kesenangan maksimum yang bisa didapatkan untuk kasus tersebut. Output Y dengan ketelitian 3 angka di belakang koma.

Contoh input	Output untuk contoh input
4 2 10 2 1 1 5 4 10 9 5 7 8 4 4 6 5 1 2 3 2 1 3 3 20 6 19 5 5 18 18	Kasus #1: 4.500 Kasus #2: 0.000 Kasus #3: 6.000 Kasus #4: 10.200

Penjelasan contoh kasus 1

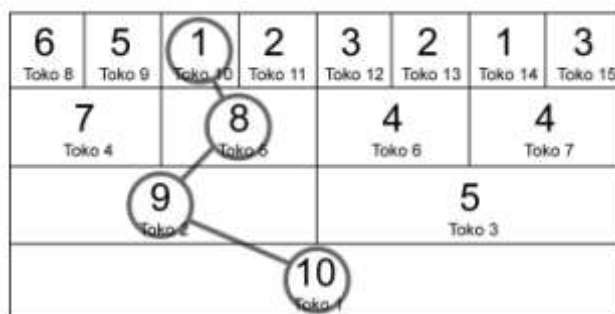
Rute perjalanan yang menghasilkan nilai kesenangan tertinggi adalah: Toko 1, Toko 3. Rata-rata harga baju yang Panci lihat adalah $(10 + 1) / 2 = 5.5$ dan Panci membeli baju dengan harga 1, sehingga nilai kesenangan Panci adalah $5.5 - 1 = 4.5$.

Penjelasan contoh kasus 2

Hanya ada satu toko yang bisa dikunjungi.

Penjelasan contoh kasus 3

Rute perjalanan yang menghasilkan nilai kesenangan tertinggi adalah: Toko 1, Toko 2, Toko 5, dan Toko 10. Rata-rata harga baju yang Panci lihat adalah $(10 + 9 + 8 + 1) / 4 = 7$ dan Panci membeli baju dengan harga 1, sehingga nilai kesenangan Panci adalah $7 - 1 = 6$.



Penjelasan contoh kasus 4

Toko-toko yang harus dikunjungi untuk menghasilkan nilai kesenangan tertinggi adalah Toko 1, 2, 3, 6 dan 7. Salah satu rute yang memungkinkan adalah: Toko 2 - 1 - 3 - 6 - 3 - 7 (perhatikan: toko 3 dikunjungi dua kali, namun harga bajunya tetap hanya akan dihitung satu kali). Rata-rata harga baju yang Panci lihat adalah $(20 + 6 + 19 + 18 + 18) / 5 = 16.2$ dan Panci membeli baju dengan harga 6, sehingga nilai kesenangan Panci adalah $16.2 - 6 = 10.2$.



Problem C Bensin Cinta

Saya memiliki seorang kekasih yang berprofesi sebagai penjual bensin eceran. Belakangan ini hubungan kami kurang baik dan ia hampir saja mengakhiri hubungan kami. Ia selalu memberi saya sebuah tugas, yang jika bisa saya selesaikan dengan baik maka hubungan kami akan terus berlanjut. Ya, saya memang terlihat hanya dimanfaatkan olehnya, tapi saya rela melakukan ini semua demi dia. Itulah sebabnya saya selalu berusaha keras dalam menyelesaikan setiap tugas yang dia berikan. Namun rupa-rupanya tugas kali ini membuat saya cemas sehingga saya memerlukan bantuan Anda untuk menyelesaikannya.

Berikut adalah tugas yang diberikan. Terdapat sebuah kendaraan bermotor yang mula-mula berisi A unit bensin dengan kapasitas tangki sebesar M unit di rumah kekasih saya. Kendaraan ini memerlukan C unit bensin untuk bergerak sejauh 1 km. Ia meminta saya untuk mengisi galon-galon di rumahnya yang saat ini kosong dengan K unit bensin. Saya harus menggunakan kendaraan ini untuk pergi ke pom bensin terdekat yang berjarak 1 km dari rumahnya dan mengisi tangki kendaraan ini, dan kemudian pulang ke rumahnya dan memindahkan isi tangki ke dalam galon. Saya boleh bolak-balik antara rumahnya dan pom bensin berapa kalipun yang diperlukan selama total jarak yang saya tempuh tidak lebih dari B km. Bisakah saya menyelesaikan tugas yang diberikan?

* Saya tidak boleh membawa galon tersebut ke pom bensin karena saya akan terjerat oleh UU Migas Nomor 22/2001 pasal 53 berupa pidana tiga tahun penjara dan denda maksimal Rp.30.000.000.000,-

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 1.000$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari lima buah bilangan bulat A, M, C, K dan B dalam satu baris secara berurutan sesuai dengan deskripsi permasalahan di atas. Batasan untuk masing-masing input adalah:

- $1 \leq M, C, K \leq 10^9$
- $1 \leq A \leq M$
- $1 \leq B \leq 2 * 10^9$

Output

Untuk setiap kasus, output dalam satu baris "Kasus #X: Y" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1, dan Y adalah total jarak yang harus saya tempuh untuk menyelesaikan tugas yang diberikan. Apabila saya tidak bisa menyelesaikan tugas yang diberikan, maka output "putus" (tanpa kutip) untuk Y.

Contoh input	Output untuk contoh input
4 1 6 1 7 4 3 9 2 8 4 1 6 1 7 3 3 10 4 2 6	Kasus #1: 4 Kasus #2: 2 Kasus #3: putus Kasus #4: 0



Penjelasan contoh kasus 1

- (di rumah) mula-mula tangki berisi 1 unit bensin.
- Menuju pom bensin, bensin di tangki berkurang 1 unit, menjadi 0 unit.
- (di pom bensin) isi tangki menjadi 6 unit bensin.
- Menuju rumah, bensin di tangki berkurang 1 unit, menjadi 5 unit.
- (di rumah) pindahkan 4 unit bensin dari tangki ke galon, sehingga tangki tersisa 1 unit bensin.
- Menuju pom bensin, bensin di tangki berkurang 1 unit, menjadi 0 unit.
- (di pom bensin) isi tangki menjadi 6 unit bensin.
- Menuju rumah, bensin di tangki berkurang 1 unit, menjadi 5 unit.
- (di rumah) pindahkan 3 unit bensin dari tangki ke galon, sehingga tangki tersisa 2 unit bensin.

Total bensin yang berhasil dipindahkan ke galon = $4 + 3 = 7$ unit.

Total jarak yang saya tempuh = 4 km (dua kali bolak-balik).

Penjelasan contoh kasus 2

- (di rumah) mula-mula tangki berisi 3 unit bensin.
- (di rumah) pindahkan 1 unit bensin dari tangki ke galon, sehingga tangki tersisa 2 unit bensin.
- Menuju pom bensin, bensin di tangki berkurang 2 unit, menjadi 0 unit.
- (di pom bensin) isi tangki menjadi 9 unit bensin.
- Menuju rumah, bensin di tangki berkurang 2 unit, menjadi 7 unit.
- (di rumah) pindahkan 7 unit bensin dari tangki ke galon, sehingga tangki tersisa 0 unit bensin.

Total bensin yang berhasil dipindahkan ke galon = $1 + 7 = 8$ unit.

Total jarak yang saya tempuh = 2 km (satu kali bolak-balik).

Penjelasan contoh kasus 3

Tugas yang diberikan sama dengan contoh kasus 1 di atas, namun kali ini saya hanya boleh menempuh maksimal 3 km. Tidak ada cara untuk menyelesaikan tugas ini, sehingga hubungan saya pasti diakhiri olehnya.

Penjelasan contoh kasus 4

Saya cukup memindahkan 2 unit bensin dari tangki ke galon, sehingga tangki tersisa 1 unit bensin. Galon yang ada berisi 2 unit bensin sehingga tugas saya berhasil diselesaikan tanpa perlu pergi ke pom bensin.



Problem D

Pasangan Anagram

Istilah “string” dalam pemrograman komputer biasanya mengacu pada deret simbol atau barisan karakter. Berbeda dengan “kata”, string tidak harus memiliki semantik (arti), contoh: XZAAOP adalah sebuah string; namun tidak ada kata XZAAOP karena XZAAOP tidak memiliki arti, setidaknya di dalam Bahasa Indonesia yang kita ketahui.

Sebuah string A disebut sebagai anagram dari string B jika dan hanya jika huruf-huruf penyusun A bisa diacak dan diatur sedemikian rupa membentuk string B. Contoh: DATA dan ADAT adalah pasangan anagram, begitu juga dengan LAMA dan AMAL, namun API dan APA bukanlah pasangan anagram karena huruf-huruf penyusun API tidak bisa membentuk string APA.

Diberikan sebuah kamus yang berisi N buah string, tugas Anda adalah menghitung ada berapa pasangan anagram yang bisa dijumpai di dalam kamus tersebut. Catatan: setiap pasangan anagram terdiri dari tepat dua string. Di dalam kamus tersebut bisa saja dijumpai dua atau lebih string yang sama, namun mereka harus diperlakukan layaknya string yang berbeda.

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 ($1 \leq N \leq 5.000$) yang menyatakan banyaknya string di dalam kamus yang diberikan. N baris berikutnya masing-masing terdiri dari sebuah string S yang disusun oleh huruf kapital A..Z. Panjang S adalah antara 1 hingga 10, inklusif.

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 pasangan anagram yang bisa dijumpai dalam kamus yang diberikan pada kasus tersebut.

Contoh input	Output untuk contoh input
4 4 DATA ADAT TADA TAAD 3 CC ABC CBC 5 BINUS BNPCHS INBUS CHAMPION CSPNBH 2 ABCDEFGHIJ ABCDEFGHIJ	Kasus #1: 6 Kasus #2: 0 Kasus #3: 2 Kasus #4: 1



Penjelasan contoh kasus 1

Keempat string di dalam kamus yang diberikan adalah anagram satu dengan yang lainnya. Dengan demikian, kita bisa menjumpai 6 pasangan anagram pada kasus ini:

1. DATA dan ADAT
2. DATA dan TADA
3. DATA dan TAAD
4. ADAT dan TADA
5. ADAT dan TAAD
6. TADA dan TAAD

Penjelasan contoh kasus 2

Tidak ada pasangan anagram yang bisa dijumpai di kasus ini.

Penjelasan contoh kasus 3

Dua pasangan anagram yang bisa dijumpai adalah:

1. BINUS dan INBUS
2. BNPCHS dan CSPNBH

Penjelasan contoh kasus 4

Satu pasangan anagram yang bisa dijumpai adalah:

1. ABCDEFGHIJ dan ABCDEFGHIJ

Kedua string tersebut dianggap string yang berbeda meskipun mereka sama.



Problem E

Menulis Esai

Panci sebentar lagi akan mengikuti UTS (Ujian Tengah Semester) di Universitas Bambu. Berbeda dengan ujian pada umumnya, ujian yang akan Panci ikuti berupa esai dan Panci harus menuliskan gagasan-gagasan yang sudah ia pelajari sebelumnya di dalam esai tersebut. Panci sudah mempelajari N buah gagasan yang masing-masing memerlukan durasi D_i menit untuk ditulis. Ujian kali ini hanya terdiri dari W menit dan setiap peserta ujian hanya diperbolehkan untuk menuliskan tidak lebih dari M gagasan.

Yang ingin diuji dalam ujian kali ini adalah kemampuan mahasiswa dalam menghubungkan gagasan-gagasan yang ada, sehingga setiap gagasan yang berdiri sendiri tidak memiliki nilai. Dosen yang memeriksa akan memberi nilai sebesar A_{ij} ($i \neq j$) jika gagasan ke- i dan gagasan ke- j keduanya ditulis pada lembar jawaban (catatan: $A_{ij} = A_{ji}$). Setiap nilai A_{ij} adalah sebuah bilangan 2^K dan unik, dengan kata lain, tidak ada dua nilai A_{ij} yang sama untuk setiap pasangan gagasan i dan j yang berbeda. Tentunya setiap gagasan hanya perlu dituliskan satu kali dalam lembar jawaban.

Dosen yang memeriksa ujian ini rupanya juga suka mengubah-ubah nilai A_{ij} , sehingga permasalahan ini bisa direpresentasikan dengan Q buah query. Ada 2 jenis query:

- $C\ p\ q\ r$: Nilai A_{pq} (dan A_{qp}) diubah menjadi r .
- Q : output gagasan mana saja yang harus dituliskan untuk mendapatkan nilai maksimal.

Untuk setiap query C , dijamin setiap A_{ij} yang dihasilkan tetap unik.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan empat buah bilangan bulat N , M , W , dan Q dalam satu baris secara berurutan sesuai dengan penjelasan di atas. Baris berikutnya berisi N buah bilangan bulat D_i yang menyatakan durasi penulisan gagasan ke- i secara berurutan. N baris berikutnya masing-masing berisi N bilangan bulat (membentuk matrix berukuran $N \times N$) yang menginformasikan nilai K untuk poin A_{ij} (ingat: $A_{ij} = 2^K$). Perhatikan bahwa khusus isi diagonal matrix ($i = j$) akan selalu bernilai 0 dan harus diabaikan (artinya 0 poin, bukan 2^0). Q baris berikutnya masing-masing berisi query seperti yang dijelaskan di atas. Batasan untuk masing-masing input adalah:

- $1 \leq M, p, q \leq N \leq 50$
- $1 \leq W \leq 10^6$
- $1 \leq Q \leq 100$
- $1 \leq D_i \leq 10^6$
- $1 \leq K, r \leq 10^5$

Output

Untuk setiap kasus, output dalam satu baris "Kasus #X:" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1. Untuk setiap query Q pada input, tuliskan nomor gagasan yang harus Panci tuliskan (terurut menaik, masing-masing dipisahkan oleh tepat sebuah spasi) untuk mendapatkan nilai maksimum. Asumsikan selalu ada minimal satu gagasan yang bisa ditulis. Jika ada lebih dari satu kemungkinan jawaban, keluarkan jawaban yang paling kecil secara leksikografis (urutan kamus).



Contoh input	Output untuk contoh input
3 3 2 15 3 4 6 8 0 3 2 3 0 1 2 1 0 Q C 1 3 4 Q 3 1 15 1 4 6 8 0 3 2 3 0 1 2 1 0 Q 4 3 10 5 3 3 3 5 0 5 6 1 5 0 4 2 6 4 0 3 1 2 3 0 Q C 1 4 7 Q C 2 3 10 Q	Kasus #1: 1 2 1 3 Kasus #2: 1 Kasus #3: 1 2 3 1 4 1 2 3

Penjelasan contoh kasus 1

Solusi optimal untuk query pertama adalah dengan menuliskan gagasan nomor 1 dan 2 dengan total nilai $2^3 = 8$ dan waktu yang dibutuhkan adalah $4 + 6 = 10$. Setelah ada perubahan nilai pada query kedua ($A_{13} = A_{31} = 4$), maka solusi optimal untuk query ketiga adalah dengan menulis gagasan nomor 1 dan 3 dengan total nilai $2^4 = 16$ dan waktu yang dibutuhkan adalah $4 + 8 = 12$.

Penjelasan contoh kasus 2

Pada kasus ini Panci hanya diperbolehkan untuk menulis satu gagasan, sehingga gagasan manapun yang ditulis akan menghasilkan nilai akhir yang sama, yaitu 0.

Penjelasan contoh kasus 3

Query 1: tuliskan gagasan (1, 2, 3) dengan nilai $2^5 + 2^6 + 2^4 = 112$ dan total waktu $3 + 3 + 3 = 9$.

Query 3: tuliskan gagasan (1, 4) dengan nilai $2^7 = 128$ dan total waktu $3 + 5 = 8$.

Query 5: tuliskan gagasan (1, 2, 3) dengan nilai $2^5 + 2^6 + 2^{10} = 1120$ dan total waktu $3 + 3 + 3 = 9$.



Problem F Satu Nol

Pendaftaran calon mahasiswa baru untuk jurusan Teknik Informatika (CS) di Bambu University sudah dibuka. Banyak sekali pelajar-pelajar dari seluruh negeri yang mendaftar, termasuk di antaranya Putra, pemenang BPCHS (Bambu Programming Contest for High-School Student) 2013 yang lalu. Sebagai pemenang BPCHS, dia mendapatkan hak istimewa berupa pembebasan biaya masuk dan beasiswa kuliah hingga lulus.

Putra memiliki hobi mengamati fisik seseorang, terutama berat dan tinggi seseorang. Ia selalu tertawa ketika ada seorang yang kurus dan tinggi berjalan bersebelahan dengan seseorang yang gemuk dan pendek (ia selalu menghubungkan situasi ini dengan angka “10” – apapun itu artinya) ¹.

Tahun ini ada N mahasiswa baru jurusan CS yang bergabung ke Bambu University dan Putra berhasil mendapatkan data berat dan tinggi semua mahasiswa baru. Putra mendefinisikan “nilai kombinasi” dari dua orang sebagai jumlah dari selisih berat dan selisih tinggi mereka. Lebih tepatnya, untuk setiap dua orang i dan j yang berbeda dengan berat W_i dan W_j serta tinggi H_i dan H_j , nilai kombinasi mereka adalah $|W_i - W_j| + |H_i - H_j|$, di mana $|\cdot|$ adalah operator nilai absolut.

Putra ingin mengetahui nilai kombinasi tertinggi yang bisa didapatkan dari data-data yang ia miliki. Tentunya bagi Putra yang pernah menjuarai BPCHS, ini adalah hal yang mudah; ia ingin menantang Anda untuk menyelesaikan permasalahan yang sama.

¹ Jangan tiru sifat buruk Putra yang satu ini; kita tidak seharusnya mendiskriminasi orang berdasarkan fisik.

Input

Baris pertama berisi sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan sebuah bilangan N ($2 \leq N \leq 40.000$) yang menyatakan banyaknya mahasiswa baru tahun ini. N baris berikutnya masing-masing terdiri dari dua buah bilangan bulat W_i dan H_i ($1 \leq W_i, H_i \leq 1.000.000$) yang menyatakan berat dan tinggi mahasiswa ke- i secara berurutan.

Output

Untuk setiap kasus, cetak “Kasus #X: Y” dimana X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah nilai kombinasi tertinggi yang bisa didapatkan dari data-data mahasiswa baru di kasus tersebut.

Contoh input	Output untuk contoh input
2 5 1 1 1 2 2 1 5 5 5 1 2 3 3 3 3	Kasus #1: 8 Kasus #2: 0



BNPC

The 2014
Bina Nusantara Programming Contest
for High School Students



Penjelasan contoh kasus 1

Nilai kombinasi tertinggi didapatkan dari pasangan mahasiswa 1 (dengan berat 1 dan tinggi 1) dan mahasiswa 4 (dengan berat 5 dan tinggi 5). Nilai kombinasinya adalah $|1 - 5| + |1 - 5| = 8$.

Penjelasan contoh kasus 2

Hanya ada dua mahasiswa di kasus ini dan keduanya memiliki berat dan tinggi yang identik. Sehingga di kasus ini, nilai kombinasi tertinggi yang bisa didapatkan adalah 0.



Problem G Gedung Baru

Vincent adalah seorang kuli bangunan penuh talenta yang diminta untuk membangun gedung kelima BINUS University di Jawa Tengah. Untuk mempermudah permasalahan, mari kita asumsikan Jawa Tengah terbentang dari kiri ke kanan (hanya 1 dimensi) dengan ketinggian tanah untuk setiap meter ke- i dari kiri adalah H_i meter di atas laut dan total lebar Jawa Tengah adalah M meter. Gedung yang ingin dibangun oleh BINUS memiliki lebar N meter.

Tentunya Vincent memahami bahwa gedung ini harus dibangun di atas permukaan tanah sedemikian sehingga gedung ini bisa berdiri tegak sejajar dengan permukaan air laut. Dengan talentanya, Vincent mampu membangun gedung yang hanya membutuhkan dua hal berikut:

- Minimal 1 meter tanah yang menopang $N/2$ meter bagian kiri gedung.
- Minimal 1 meter tanah yang menopang $N/2$ meter bagian kanan gedung.

Gedung yang dibangun bisa tetap berdiri meskipun hanya ditopang oleh 2 meter tanah seperti yang dijelaskan di atas. Tentunya ketinggian tanah yang menopang gedung ini harus sama agar bangunan ini bisa berdiri tegak sejajar dengan permukaan air laut.

Diberikan denah ketinggian tanah di Jawa Tengah selebar M meter, tentukan ada berapa lokasi yang mungkin digunakan untuk membangun gedung baru BINUS yang memiliki lebar N meter (tanpa memanipulasi ketinggian tanah). Semua bagian gedung baru BINUS harus berada di wilayah Jawa Tengah; tidak boleh ada bagian dari gedung yang berada di luar Jawa Tengah.

Input

Baris pertama berisi sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan dua buah bilangan bulat M dan N ($2 \leq N \leq M \leq 40.000$) yang menyatakan lebar Jawa Tengah dan lebar gedung baru BINUS secara berurutan. Baris berikutnya berisi M buah bilangan bulat H_i ($1 \leq H_i \leq 4.000$) yang merepresentasikan ketinggian tanah pada meter ke- i dari kiri secara berurutan. Untuk mempermudah permasalahan, asumsikan N adalah bilangan genap.

Output

Untuk setiap kasus, cetak "Kasus #X: Y" dimana X adalah nomor kasus dimulai dari 1 secara berurutan, dan Y adalah banyaknya lokasi yang bisa digunakan untuk membangun gedung baru BINUS pada kasus tersebut.

Contoh input	Output untuk contoh input
4 10 4 1 2 3 2 3 3 5 5 5 10 4 2 1 1 1 1 5 4 10 15 12 10 9 6 4 100 50 100 100 50 50	Kasus #1: 3 Kasus #2: 3 Kasus #3: 0 Kasus #4: 2



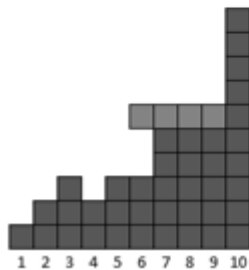
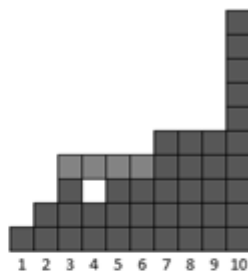
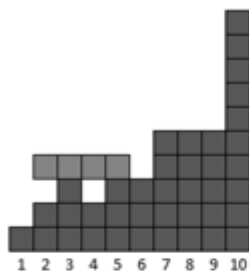
Penjelasan contoh kasus 1

Tiga gambar di bawah merepresentasikan lokasi-lokasi yang bisa digunakan untuk membangun gedung BINUS selebar 4 meter pada kasus ini.

#	Lokasi	Bagian Kiri ¹	Bagian Kanan ²	Ketinggian Tanah
1	[2, 5]	3	5	3
2	[3, 6]	3	5, 6	3
3	[6, 9]	7	8, 9	5

¹ N/2 bagian kiri gedung ditopang oleh tanah pada meter ke ...

² N/2 bagian kanan gedung ditopang oleh tanah pada meter ke ...





Problem H

Benteng Takeshi

Benteng Takeshi adalah sebuah program televisi yang dahulu cukup populer di mana peserta saling berlomba untuk memperebutkan hadiah.

Permainan kali ini menggunakan $N \times M$ tongkat lurus dengan masing-masing tongkat berukuran 1×2 (atau 2×1 , tergantung orientasi). Tongkat-tongkat tersebut disusun pada permukaan sebuah kolam sedemikian sehingga jika dilihat dari atas akan terlihat N baris dan M kolom tongkat. Setiap tongkat dinomori dari 1 hingga $N \times M$ dari kiri ke kanan dan atas ke bawah (lihat penjelasan contoh kasus 1 di bawah untuk lebih jelasnya). Beberapa informasi penting dalam permainan ini:

1. Setiap tongkat memiliki skor yang bisa dikumpulkan oleh pemain.
2. Semua tongkat memiliki panjang 2 petak dan memiliki poros putar tepat di tengah tongkat, sehingga tongkat tersebut bisa diputar dari 1×2 menjadi 2×1 dan sebaliknya.
3. Jarak antara titik tengah tongkat (poros putar) dengan titik tengah tongkat terdekat lainnya di sisi utara, timur, selatan, dan barat adalah 3 petak; sedangkan jarak titik tengah tongkat pinggir dengan tepian kolam adalah 2 petak.
4. Skor tongkat dapat diambil setiap kali pemain mendarat di tongkat tersebut.
5. Kedua ujung tongkat berbentuk mata panah dan selalu menunjuk ke titik tengah tongkat lain.

Aturan dalam permainan ini adalah:

1. Pada saat permainan dimulai, pemain harus memilih satu tongkat yang akan ia injak untuk pertama kalinya (boleh tongkat yang mana saja, tidak harus di pinggir). Pemain akan langsung mendapatkan skor pada tongkat tersebut. Aktivitas ini memerlukan 0 energi.
2. Pemain diperbolehkan berjalan di sepanjang tongkat. Aktivitas ini memerlukan 0 energi.
3. Pemain dapat memutar tongkat tempat ia berdiri sebesar 90° (arah utara-selatan menjadi timur-barat, atau sebaliknya). Aktivitas ini memerlukan P energi. Saat pemain meninggalkan tongkat yang baru saja ia putar, tongkat ini akan kembali ke posisi semula sebelum diputar.
4. Pemain dapat melompat hanya ke tongkat lain yang bersebelahan atau ke tepian kolam sesuai dengan arah yang ditunjuk oleh mata panah tongkat tempat ia berdiri. Aktivitas ini memerlukan energi sebesar jarak lompatan yang dilakukan (dalam satuan petak).

Permainan berakhir jika:

1. Pemain berhasil melompat ke tepian kolam (skor pemain adalah semua skor tongkat yang sudah ia kumpulkan), atau
2. Pemain kehabisan energi dan masih berada di atas kolam (pemain ini dinyatakan kalah).

Jika mula-mula seorang pemain memiliki E energi, berapa total skor maksimum yang bisa ia peroleh dalam permainan ini? Tentukan juga tongkat pertama yang harus ia pilih.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan empat bilangan bulat N , M , E , dan P ($1 \leq N, M, E, P \leq 100$) dalam satu baris secara berurutan sesuai dengan deskripsi di atas. N baris berikutnya masing-masing berisi tepat M karakter '-' atau 'l' (karakter "minus" dan "L kecil") yang merepresentasikan posisi tongkat jika dilihat dari atas (1×2 atau 2×1 secara berurutan). Baris berikutnya berisi $N \times M$ bilangan bulat S_i ($0 \leq S_i \leq 100$) yang menyatakan skor tongkat ke- i ($i = 1..N \times M$).

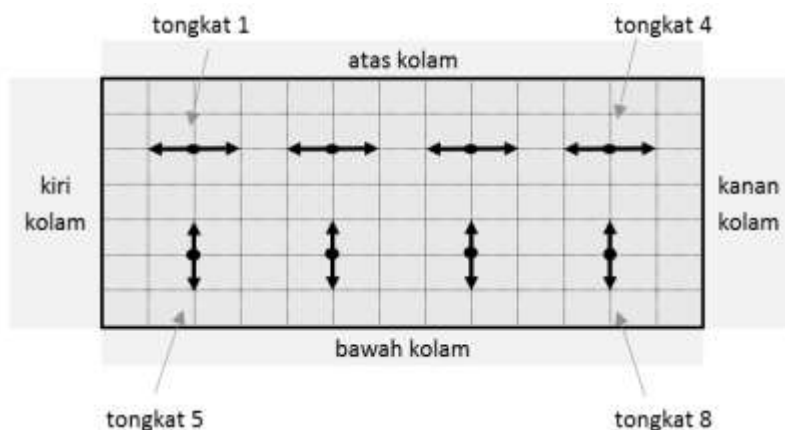


Output

Untuk setiap kasus, output dalam satu baris "Kasus #X: Y" (tanpa kutip) di mana X adalah nomor kasus dimulai dari 1. Untuk kasus dimana pemain bisa keluar ke tepian kolam, output Y terdiri dari dua bilangan bulat yang dipisahkan oleh tepat satu buah spasi: skor tertinggi yang bisa diperoleh dan nomor tongkat pertama yang harus dipilih. Jika ada lebih dari satu tongkat pertama yang bisa dipilih untuk mendapatkan skor tertinggi, output nomor tongkat yang lebih kecil. Untuk kasus dimana pemain pasti kalah, maka output "pemain pasti kalah" (tanpa kutip) untuk Y.

Contoh input	Output untuk contoh input
3 2 4 4 100 ---- 1111 0 1 2 3 10 20 30 40 2 4 8 1 ---- 1111 0 1 2 3 10 20 30 40 1 3 3 1 --- 1 1 1	Kasus #1: 43 8 Kasus #2: 113 7 Kasus #3: 3 1

Penjelasan contoh kasus 1



Tongkat pertama yang harus dipilih adalah tongkat 8, kemudian pemain melompat ke tongkat 4 dan melompat ke kanan ke tepian kolam. Total skor yang berhasil dikumpulkan adalah: $40 + 3 = 43$, dengan sisa energi: $4 - 2 - 1 = 1$.

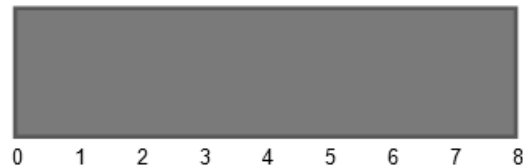
Penjelasan contoh kasus 2

Kondisi permainan sama dengan kasus 1 di atas, namun energi yang diperlukan untuk memutar tongkat di kasus ini adalah 1. Tongkat pertama yang harus dipilih adalah tongkat 7, kemudian tongkat ini diputar dan pemain melompat ke tongkat 8. Dari tongkat 8, ia melompat ke tongkat 4 dan memutar tongkat 4 untuk kembali melompat ke tongkat 8. Ia akhirnya keluar dari kolam dengan melompat ke tepian kolam yang ada di selatan. Total skor yang berhasil dikumpulkan adalah: $30 + 40 + 3 + 40 = 113$, dengan sisa energi: $8 - 1 - 2 - 2 - 1 - 1 - 1 = 0$.



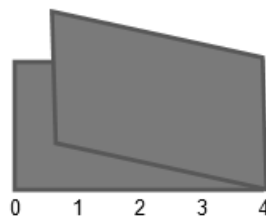
Problem I Melipat Kertas

Diberikan selembar kertas dengan panjang N cm yang lurus tanpa ada lipatan, kalian diminta untuk melipat kertas tersebut pada cm tertentu. Gambar 1 di bawah mengilustrasikan selembar kertas dengan panjang 8 cm (titik paling kiri adalah cm ke 0 dan titik paling kanan adalah cm ke 8).



Gambar 1. Kertas dengan panjang 8 cm.

Untuk setiap permintaan melipat kertas, kalian akan diberikan sebuah bilangan bulat P_i yang menyatakan titik lipatan yang diinginkan. Kalian harus melipat kertas tersebut secara horizontal ke arah kiri dengan titik lipatan tepat pada cm ke P_i . Anda boleh mengasumsikan perintah lipatan tidak akan menyebabkan ada bagian dari kertas yang berada di cm ke < 0 , atau dengan kata lain, jika panjang kertas saat ini adalah L , maka dijamin $L/2 \leq P_i < L$. Contoh, jika $L = 8$ maka berlaku $4 \leq P_i < 8$; jika $L = 7$ maka berlaku $3.5 \leq P_i < 7$.



Gambar 2. Kertas pada Gambar 1 dilipat pada titik lipatan 4.

Gambar 2 mengilustrasikan kertas Gambar 1 yang dilipat pada titik lipatan $P_i = 4$. Lipatan ini akan menghasilkan “selembar” kertas dengan panjang 4 cm (dari cm ke 0 hingga cm ke 4), sehingga jika ada perintah lipatan berikutnya, maka P_i hanya akan berada di rentang $2 \dots 3$.

Setelah kalian selesai melakukan perintah-perintah lipatan yang diberikan, buka kembali kertas yang dilipat tersebut dan perhatikan bekas lipatan yang terjadi. Ada berapa jumlah lipatan yang terbentuk pada kertas tersebut? Abaikan batasan fisik seperti ketebalan kertas pada permasalahan ini, dan asumsikan kalian mampu menyelesaikan perintah yang diberikan dengan sempurna, dengan kata lain, kalian bisa melipat kertas tersebut tepat pada titik lipatan yang ditentukan.

Input

Baris pertama dari input adalah 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 ($2 \leq N \leq 40.000$; $1 \leq M < N$) yang menyatakan panjang kertas mula-mula dan banyaknya perintah yang akan diberikan. Baris berikutnya terdiri dari M bilangan bulat P_i yang menyatakan posisi titik lipatan yang harus dilakukan. Dijamin untuk setiap perintah lipatan berlaku $L/2 \leq P_i < L$, dengan L adalah panjang kertas tepat sebelum perintah ke- i dijalankan.



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 bekas lipatan yang terbentuk setelah kalian melakukan M perintah pada kasus tersebut.

Contoh input	Output untuk contoh input
4 8 1 4 8 2 4 2 100 3 80 50 25 100 2 80 50	Kasus #1: 1 Kasus #2: 3 Kasus #3: 5 Kasus #4: 2

Penjelasan contoh kasus 1

Lipatan terbentuk pada cm ke 4 dari kertas mula-mula.

Penjelasan contoh kasus 2

Lipatan terbentuk pada cm ke 2, 4, dan 6 dari kertas mula-mula.

Penjelasan contoh kasus 3

Lipatan terbentuk pada cm ke 25, 50, 75, 80, dan 85 dari kertas mula-mula.

Penjelasan contoh kasus 4

Lipatan terbentuk pada cm ke 50 dan 80 dari kertas mula-mula.



Problem J

Wakka dan Perpustakaan Raksasa

Panda Land baru saja membuka sebuah perpustakaan umum yang sangat besar dengan koleksi buku yang super lengkap. Perpustakaan ini memiliki satu rak buku yang saaaangat panjang dan saaaangat tinggi. Rak buku ini biasanya sangat merepotkan pustakawan yang bertugas, karena mereka harus menggeser-geser tangga cukup sering (dan jauh), dan terkadang roda tangga ini perlu diberi pelumas terlebih dahulu sebelum bisa digeser-geser; jika kamu tidak tahu, tangga ini diperlukan untuk mencapai bagian-bagian rak yang tinggi.

Wakka, manusia unik, baru saja diterima bekerja sebagai pustakawan di perpustakaan tersebut. Konon Wakka memiliki relasi dengan Elastigirl (Mrs. Incredible) yang memiliki kemampuan untuk mengubah bentuk tubuhnya, termasuk memanjangkan tangannya. Wakka juga memiliki kemampuan yang serupa, namun ia hanya bisa meninggikan badan dan memanjangkan tangannya saja. Wakka memerlukan energi sebesar 1 unit ketika meninggikan badan untuk setiap meternya. Begitu juga ketika ia memanjangkan tangan, energi yang diperlukan adalah 1 unit per meter. Namun Wakka tidak bisa mengangkat tangannya ke atas ketika dipanjangkan, dengan kata lain, ia hanya bisa memanjangkan tangannya ketika tangannya mendatar – sejajar dengan sumbu x. Waka tidak memerlukan energi untuk memendekkan tangan ataupun mengembalikan badannya ke ukuran semula.

Tugas pertama Wakka sebagai pustakawan adalah mengambil N buah buku yang masing-masing berada pada posisi (x_i, y_i) – x_i meter dari tepi kiri dan y_i meter dari lantai – pada rak buku super panjang dan super tinggi tersebut. Wakka bisa meninggikan badan maupun memanjangkan tangannya untuk mengambil semua buku yang diperlukan, namun ia hanya bisa mengambil dan memegang paling banyak satu buku setiap saatnya, sehingga ia perlu meletakkan dulu buku yang baru ia ambil ke keranjang yang ia bawa sebelum mengambil buku berikutnya. Wakka tidak pernah menggunakan tangga, dan asumsikan tinggi badan (dan tangan) Wakka mula-mula adalah 0 meter (tepat di lantai), dan panjang tangan Wakka mula-mula adalah 0 meter.

Bukan rahasia lagi, Wakka juga terkenal sebagai pemalas kelas berat. Ia tidak ingin banyak berjalan menggunakan kakinya untuk mencapai buku yang diperlukan, sehingga ia akan memilih satu posisi $(K, 0)$, berdiri di sana, dan mengambil semua buku yang diperlukan hanya dengan meninggikan badan dan memanjangkan tangan saja. Wakka ingin mengetahui, jika ia memilih posisi K dengan benar, berapa total energi minimum yang ia perlu keluarkan untuk mengambil semua buku yang diperlukan?

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 ($1 \leq N \leq 20.000$) dalam satu baris yang menyatakan banyaknya buku yang perlu diambil. N baris berikutnya masing-masing terdiri dari dua buah bilangan bulat x_i dan y_i ($1 \leq x_i, y_i \leq 20.000$) yang menyatakan posisi buku ke- i di rak tersebut.



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 energi minimal yang diperlukan Wakka untuk mengambil semua buku jika ia memilih posisi (K, 0) dengan benar untuk kasus tersebut.

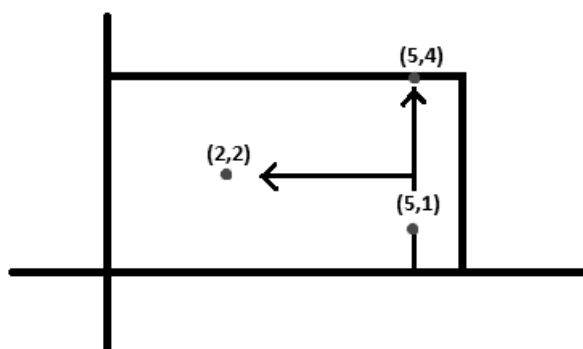
Contoh input	Output untuk contoh input
3 3 2 2 5 1 5 4 5 1 1 2 2 3 3 4 4 5 5 2 6 4 10 12	Kasus #1: 7 Kasus #2: 11 Kasus #3: 16

Penjelasan contoh kasus 1

Ada 3 buku dengan masing-masing berada pada posisi: (2, 2), (5, 1), dan (5, 4). Untuk meminimalkan total energi yang diperlukan, Wakka bisa memilih $K = 5$ dan berdiri di posisi (5, 0), kemudian:

- Tinggikan badan 1 meter dan ambil buku di posisi (5, 1), energi = 1.
- Tinggikan badan 1 meter lagi, energi = 1.
- Panjangkan tangan ke kiri dan ambil buku di posisi (2, 2), energi = 3.
- Tinggikan badan 2 meter dan ambil buku di posisi (5, 4), energi = 2.

Total energi yang diperlukan dengan cara ini adalah $1 + 1 + 3 + 2 = 7$.



Penjelasan contoh kasus 2

Wakka bisa memilih $K = 3$ untuk meminimalkan total energi yang diperlukan.

Penjelasan contoh kasus 4

Wakka bisa memilih $K = 7$ untuk meminimalkan total energi yang diperlukan. Tentunya ada posisi K lain yang juga bisa meminimalkan total energi yang diperlukan pada contoh kasus ini.