



## Problem A

### 3 Prima

Seperti yang telah anda ketahui tentunya, bilangan prima adalah bilangan bulat positif yang memiliki tepat dua pembagi, yaitu 1 dan bilangan itu sendiri. Lima bilangan prima paling kecil adalah 2, 3, 5, 7 dan 11. Semua bilangan lain selain bilangan prima disebut sebagai bilangan komposit, karena bilangan-bilangan tersebut bisa dibentuk dari mengalikan dua atau lebih bilangan prima.

Diberikan dua buah bilangan bulat A dan B dengan  $A \leq B$ , tugas anda adalah menghitung ada berapa banyak bilangan di antara A dan B, inklusif, yang bisa dibentuk dengan mengalikan tepat tiga bilangan prima yang berbeda. Contoh bilangan yang demikian adalah, 30 ( $2 * 3 * 5$ ), 357 ( $3 * 7 * 17$ ), 11687 ( $13 * 29 * 31$ ). Sedangkan bilangan seperti 75 ( $3 * 5 * 5$ ) bukanlah bilangan yang dicari karena 75 tidak dibentuk dari perkalian tiga bilangan prima yang berbeda.

#### Input

Baris pertama dari input adalah sebuah bilangan bulat T ( $T \leq 10000$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari dua buah bilangan bulat A dan B ( $1 \leq A \leq B \leq 5000000$ ).

#### 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 bilangan dari A hingga B inklusif yang bisa dibentuk dengan mengalikan tepat tiga bilangan prima yang berbeda.

Contoh input	Output untuk contoh input
4 1 50 60 69 1 1000 1000 1100	Kasus #1: 2 Kasus #2: 1 Kasus #3: 135 Kasus #4: 20

#### Penjelasan contoh input 1

Bilangan yang demikian di antara 1..50 adalah 30 ( $2 * 3 * 5$ ) dan 42 ( $2 * 3 * 7$ ).

#### Penjelasan contoh input 2

Bilangan yang demikian di antara 60..69 adalah 66 ( $2 * 3 * 11$ ).

## Problem B

### Kodok Melompat

Seekor kodok sedang berada di tepian sungai dan ingin melompat ke sisi sungai yang ada di seberangnya. Sungai tersebut memiliki lebar sebesar  $L$  dan di sungai tersebut terdapat  $N$  buah bilah kayu yang sedang hanyut terbawa air. Bilah-bilah kayu ini bergerak dengan kecepatan tetap 1 unit per detik dengan arah horizontal (bisa ke kiri ataupun ke kanan).

Kita asumsikan kodok tersebut berada pada koordinat  $(0, 0)$  dan ingin menyeberang ke koordinat  $(0, L+1)$ . Kodok tersebut hanya mau melompat lurus ke depan dan sekali ia melompat, ia akan terus melompat setiap detiknya (ia memerlukan satu detik untuk melompat dari satu bilah kayu ke bilah kayu lainnya yang ada tepat di depannya) hingga sampai di tepian sungai yang ada di seberang.

Meskipun kodok adalah binatang amfibi yang bisa hidup di dua alam, kodok yang satu ini takut dengan air, ia tidak ingin tercebur ke sungai ketika dia menyeberangi sungai tersebut. Oleh karena itu, ia harus memanfaatkan bilah-bilah kayu yang hanyut tersebut sebagai tempat pijakannya. Sang kodok tidak tercebur jika dan hanya jika ketika ia melompat ke petak baru (koordinat  $<0, y+1>$ ) dan di petak tersebut terdapat bilah kayu.

Untuk lebih jelasnya, anda bisa melihat penjelasan contoh kasus 1 di akhir soal ini.

Diberikan lebar sungai dan posisi serta arah semua bilah kayu yang ada, tentukan waktu tercepat di mana sang kodok bisa mencapai tepian sungai di sisi seberang. Output -1 jika sang kodok tidak akan pernah bisa mencapai tepian sungai seberang tanpa tercebur.

#### Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 20$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan dua buah bilangan bulat  $L$  dan  $N$  ( $1 \leq L \leq 1000$ ;  $1 \leq N \leq 50000$ ) yang menyatakan lebar sungai dan jumlah bilah kayu yang ada.  $N$  baris berikutnya masing-masing terdiri dari 4 bilangan bulat  $D, R, A, B$  ( $D \in \{0, 1\}$ ;  $1 \leq R \leq L$ ;  $-10^9 \leq A \leq B \leq 10^9$ ).  $D$  menyatakan arah pergerakan dari bilah kayu, 0 artinya bilah kayu tersebut bergerak dari kiri ke kanan, sedangkan 1 artinya bilah kayu tersebut bergerak dari kanan ke kiri.  $R$  menyatakan koordinat  $y$  dari bilah kayu tersebut, sedangkan  $A$  dan  $B$  menyatakan segment  $x$  dari bilah kayu. Dengan kata lain, terdapat sebuah bilah kayu pada koordinat  $(A, R)$  hingga  $(B, R)$  yang sedang bergerak sesuai arah  $D$ . Asumsikan tidak ada bilah kayu yang saling berpotongan (namun bisa saja ada bilah kayu yang bersinggungan) dan semua bilah kayu pada koordinat  $y$  yang sama selalu bergerak ke arah yang sama.

#### 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 waktu tercepat yang diperlukan sang kodok untuk mencapai sisi seberang. Output  $Y = -1$  jika sang kodok tidak mungkin mencapai sisi seberang tanpa tercebur.



# BNPC

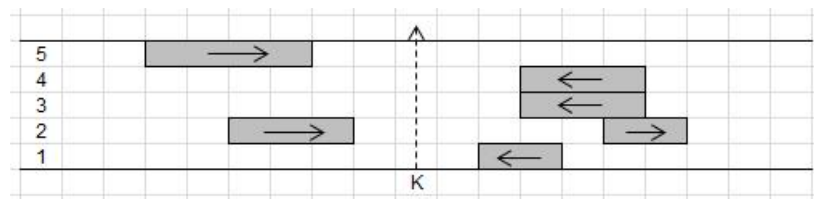
The 2012  
Bina Nusantara Programming Contest  
for High School Students



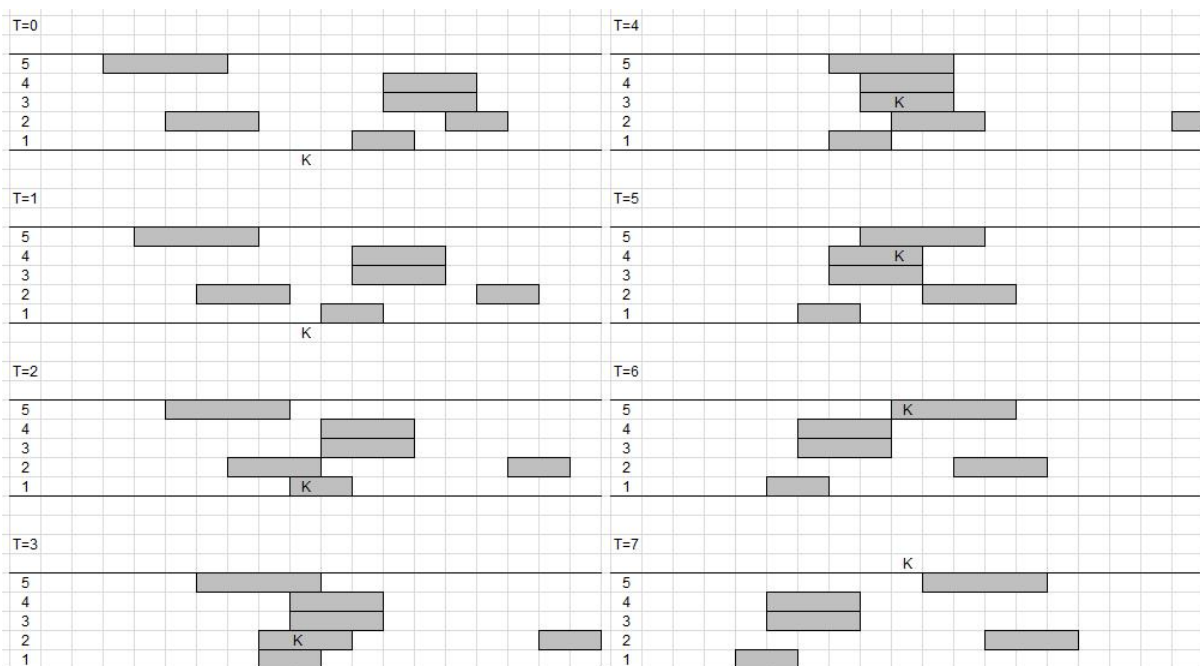
Contoh input	Output untuk contoh input
3 5 6 1 1 2 3 0 2 -4 -2 0 2 5 6 1 3 3 5 1 4 3 5 0 5 -6 -3 2 4 1 1 100 1005 0 2 -95 -90 0 2 -10 10 1 1 30 50 1 3 1 1 10000 10010 1 1 507 507 1 1 900 1000	Kasus #1: 7 Kasus #2: -1 Kasus #3: 508

### Penjelasan contoh kasus 1

Berikut ini adalah situasi pada contoh kasus 1:



Pada contoh ini, sang kodok harus mulai melompat setelah detik ke 1 dan mendarat di bilah kayu pada detik ke 2 (begitu melompat, di detik berikutnya ia juga harus segera melompat terus hingga mencapai sisi seberang) dan akan tiba di sisi sungai yang di seberang pada detik ke 7.



## Problem C

### Naskah Kuno

Sebuah kata atau frasa disebut palindrom apabila kata tersebut dapat dibaca dengan sama, baik dari depan ataupun dari belakang. Contoh: A, BB, RADAR, KODOK dan TAAT adalah palindrom, sedangkan ADAM, MINUM dan API bukanlah palindrom.

Sulaiman, seorang arkeolog Indonesia, menemukan sebuah naskah kuno dari jaman kerajaan Sriwijaya. Uniknya naskah ini tidak ditulis menggunakan bahasa Sansekerta ataupun Kawi seperti naskah-naskah lainnya di jaman yang sama, sehingga Sulaiman kesulitan untuk memahami isi dan arti dari naskah tersebut. Satu hal yang Sulaiman ketahui adalah, konon penduduk di daerah tempat naskah tersebut ditemukan selalu menggunakan kata-kata dalam bentuk palindrom dalam bahasa mereka.

Untuk memahami naskah tersebut, Sulaiman harus terlebih dahulu mencari tahu tepatnya berapa kata yang menyusun naskah tersebut. Setiap simbol pada naskah tersebut ditulis secara berurutan tanpa ada jeda atau spasi pemisah antar kata. Untungnya simbol-simbol yang digunakan pada naskah tidak banyak sehingga setiap simbol bisa dipetakan ke salah satu karakter pada alphabet. Contoh. ADAMAKAMKODOK tersusun atas tiga buah kata palindrom: ADA, MAKAM dan KODOK. ABA tersusun atas satu buah kata palindrom: ABA, tentunya kata ini juga bisa disusun dari tiga buah kata palindrom: A, B dan A, namun untuk permasalahan ini, Sulaiman menginginkan jumlah kata penyusun yang paling sedikit.

Diberikan string yang merupakan isi naskah tersebut, hitung berapa jumlah kata minimum penyusun string tersebut. Ingat, setiap kata penyusun haruslah berupa palindrom. Catatan: jika ada kata yang digunakan lebih dari satu kali, maka kata tersebut dihitung sejumlah banyak penggunaan kata tersebut.

#### Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 100$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri atas sebuah string  $S$  dalam satu baris yang panjangnya antara 1 hingga 1000, inklusif.  $S$  hanya disusun atas karakter A..Z.

#### 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 sebuah bilangan bulat yang menyatakan jumlah minimal kata yang menyusun string input.

Contoh input	Output untuk contoh input
5	Kasus #1: 3
ADAMAKAMKODOK	Kasus #2: 1
ABA	Kasus #3: 6
PQRSTU	Kasus #4: 4
ABCBDDA	Kasus #5: 2
ABCBDDBCBATT	

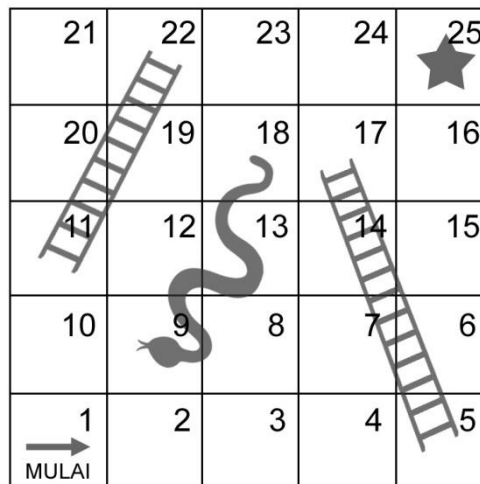


## Problem D

### Ular Tangga

Ular Tangga adalah permainan untuk anak-anak yang dimainkan oleh dua orang atau lebih. Permainan ini menggunakan sebuah papan yang dibagi menjadi kotak-kotak berukuran  $N \times N$  (biasanya  $10 \times 10$ ) dengan setiap kotak diberi nomor dari  $1..N^2$  sedemikian sehingga kotak bernomor  $i$  bersebelahan dengan kotak bernomor  $i+1$ . Gambar di bawah adalah contoh papan Ular Tangga yang berukuran  $5 \times 5$ .

21	22	23	24	25
20	19	18	17	16
11	12	13	14	15
10	9	8	7	6
1 → MULAI	2	3	4	5



Permainan dimulai dengan meletakkan bidak di kotak bernomor 1. Kemudian pemain secara bergantian akan melempar dadu (dengan angka  $1..6$ ) dan melangkahkan bidaknya sesuai angka yang keluar pada dadu yang dilempar. Contoh: jika saat ini bidak pada kotak bernomor 1 dan dadu menunjukkan angka 3, maka bidak tersebut akan melangkah hingga kotak  $1 + 3 = 4$ .

Pada papan permainan juga terdapat gambar ular dan tangga yang masing-masing menghubungkan dua buah kotak yang berbeda. Jika bidak mendarat di kotak yang memiliki ujung bawah tangga, maka bidak tersebut akan langsung naik ke kotak ujung atas tangga. Pada contoh gambar di atas, jika bidak berada di kotak 5 maka ia akan dipindahkan ke kotak 17, dan jika ia berada di kotak 11 maka ia akan dipindahkan ke kotak 22. Sedangkan jika bidak berada pada ekor ular, maka bidak tersebut akan meluncur turun menuju kepala ular. Pada contoh di atas, jika bidak berada pada kotak 18 maka ia akan meluncur turun ke kotak 9.

Perhatikan bahwa bidak hanya memanjat naik atau meluncur turun jika bidak tersebut mendarat dan berhenti tepat pada kotak dengan ujung bawah tangga atau ekor ular. Contoh: jika bidak berada pada kotak 3 dan lemparan dadu menunjukkan angka 4, maka bidak tersebut akan maju ke kotak 7. Meskipun di dalam perjalanannya bidak tersebut melewati kotak 5, namun ia tidak boleh menaiki tangga tersebut karena ia tidak berhenti di kotak tersebut.

Pemain yang terlebih dahulu mencapai kotak ke  $N^2$  adalah pemenangnya. Diberikan papan permainan Ular Tangga lengkap dengan ukuran dan posisi semua ular beserta tangga, tentukan jumlah lemparan dadu minimum yang diperlukan agar bidak pemain bisa mencapai posisi kotak terakhir dari kotak nomor 1.



## 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 bilangan bulat  $N$  ( $2 \leq N \leq 100$ ) dan  $K$  yang menyatakan ukuran papan ( $N \times N$ ) dan jumlah ular/tangga secara berurutan.  $K$  baris berikutnya masing-masing terdiri dari dua buah bilangan bulat  $a$  dan  $b$  ( $1 < a, b < N^2$ ;  $a \neq b$ ). Jika  $a < b$ , maka terdapat sebuah ujung bawah tangga pada kotak  $a$  yang memiliki ujung atas di kotak  $b$ , sedangkan jika  $a > b$  maka terdapat sebuah ekor ular pada kotak  $a$  yang kepalanya berada di kotak  $b$ . Dipastikan tidak ada kotak yang memiliki lebih dari satu ular/tangga (ujung manapun).

## 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 sebuah bilangan bulat yang menyatakan jumlah lemparan dadu minimum yang diperlukan agar bidak bisa mencapai posisi kotak terakhir dari posisi kotak nomor 1. Jika konfigurasi papan tidak memungkinkan bidak bisa mencapai posisi kotak terakhir, output -1.

Contoh input	Output untuk contoh input
2 5 3 5 17 11 22 18 9 5 6 11 2 12 3 13 4 14 5 15 6 16 7	Kasus #1: 3 Kasus #2: -1

### Penjelasan contoh kasus 1

Diperlukan minimum 3 kali lemparan dadu untuk mencapai kotak 25:

- Lemparan pertama : 4. Kotak 1  $\rightarrow$  kotak 5, naik tangga ke kotak 17.
- Lemparan kedua : 6. Kotak 17  $\rightarrow$  kotak 23.
- Lemparan ketiga : 2. Kotak 23  $\rightarrow$  kotak 25.

### Penjelasan contoh kasus 2

Tidak ada cara agar bidak bisa mencapai kotak 25.



## Problem E

### Lift

Pak Nardi adalah seorang pegawai kantor biasa yang tidak istimewa sama sekali. Ia sering bangun kesiangan, sehingga hampir setiap hari ia terlambat datang ke tempat kerjanya, hingga suatu hari ia menerima surat peringatan tentang kebiasaan terlambat hadir ini dari atasannya.

Kantor tempat Pak Nardi bekerja berada di lantai  $L$  pada sebuah gedung, dan ada  $N$  buah lift yang bisa digunakan dari lantai 1 untuk mencapai lantai manapun. Pak Nardi memiliki kemampuan berhitung dan cenayang yang luar biasa. Ia bisa dengan cepat menghitung dan mengetahui lantai tujuan dari setiap pengunjung lainnya yang hendak menaiki lift yang ada. Dari kebiasaan Pak Nardi tentu kita mengetahui bahwa dia tidak akan menggunakan tangga selama masih ada lift yang bisa digunakan. Untuk mencapai kantornya dengan cepat, ia harus menentukan lift mana yang harus ia naiki dan berapa lama waktu yang diperlukan agar ia bisa mencapai kantornya.

Ada banyak pengunjung di lantai 1 dan mula-mula semua lift dalam keadaan terbuka. Untuk setiap lift ke- $i$  ada sebanyak  $M_i$  pengunjung yang hendak menaiki lift tersebut dan Pak Nardi mengetahui lantai tujuan dari setiap pengunjung yang menaiki lift manapun.

Waktu yang diperlukan untuk masuk atau keluar dari lift adalah 2 detik per orang. Asumsikan semua pengunjung termasuk Pak Nardi adalah orang yang teratur dan akan antri bergantian untuk masuk atau keluar dari lift. Waktu yang diperlukan untuk lift agar membuka atau menutup pintunya adalah 2.5 detik, dan waktu yang diperlukan lift untuk naik adalah 1 detik per lantai.

Jika ada pengunjung lain yang memiliki tujuan lantai yang sama di dalam lift yang ia naiki, maka Pak Nardi selalu menjadi orang yang pertama keluar dari lift. Asumsikan semua pengunjung menaiki lift dari lantai 1 (tidak ada pengunjung lain yang naik dari lantai lain) dan semua lift bisa mengangkut berapapun pengunjung sekaligus.

Tugas anda adalah menentukan lift mana yang ia harus naiki sehingga ia bisa mencapai kantornya secepat mungkin (keluar dari lift pada lantai tempat kantornya berada), dan hitung juga berapa lama waktu yang ia perlukan. Jika ada lebih dari satu lift yang bisa mencapai lantai tujuan Pak Nardi dengan waktu yang sama, selalu pilih lift tercepat dengan nomor lift yang paling kecil.

#### Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 100$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus diawali oleh dua buah bilangan  $N$  ( $1 \leq N \leq 20$ ) dan  $L$  ( $1 \leq L \leq 50$ ), yang menyatakan banyaknya lift yang tersedia dan lantai tempat kantor Pak Nardi berada.  $N$  baris berikutnya masing-masing dimulai dengan sebuah bilangan bulat  $M_i$  ( $1 \leq M_i \leq 20$ ) yang menyatakan banyaknya pengunjung yang hendak menaiki lift ke- $i$ .  $M_i$  bilangan berikutnya di baris yang sama menyatakan lantai tujuan setiap pengunjung  $A_i$  ( $2 \leq A_i \leq 50$ ).

#### Output

Untuk setiap kasus, output "Kasus #X: A B" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan, A adalah nomor lift yang harus dinaiki Pak Nardi agar ia bisa tiba di kantornya secepat mungkin, dan B adalah waktu yang diperlukan Pak Nardi untuk mencapai kantornya secepat mungkin (keluar dari lift pada lantai tempat kantornya berada).





# BNPC

The 2012  
Bina Nusantara Programming Contest  
for High School Students



Contoh input	Output untuk contoh input
3 3 10 5 2 8 2 2 2 3 12 10 11 4 7 7 7 6 3 5 5 2 2 2 2 2 4 2 2 2 3 3 2 2 5 2 10 2 4 4 2 9 9	Kasus #1: 2 24 Kasus #2: 3 28 Kasus #3: 1 31

### Penjelasan contoh input 1

Jika Pak Nardi naik lift-1, maka waktu yang diperlukan adalah:

- 5 orang pengunjung dan Pak Nardi menaiki lift =  $(5 + 1) * 2$  = 12 detik
  - Pintu lift menutup = 2.5 detik
  - Lift dari lantai 1 naik ke lantai 2 = 1 detik
  - Pintu lift membuka = 2.5 detik
  - 4 orang keluar dari lift =  $4 * 2$  = 8 detik
  - Pintu lift menutup = 2.5 detik
  - Lift dari lantai 2 naik ke lantai 8 = 6 detik
  - Pintu lift membuka = 2.5 detik
  - 1 orang keluar = 2 detik
  - Pintu lift menutup = 2.5 detik
  - Lift dari lantai 8 naik ke lantai 10 = 2 detik
  - Pintu lift membuka = 2.5 detik
  - Pak Nardi keluar = 2 detik
- total waktu 46 detik

Sedangkan jika Pak Nardi naik lift-2, maka waktu yang diperlukan adalah:

- 3 orang pengunjung dan Pak Nardi menaiki lift =  $(3 + 1) * 2$  = 8 detik
  - Pintu lift menutup = 2.5 detik
  - Lift dari lantai 1 naik ke lantai 10 = 9 detik
  - Pintu lift membuka = 2.5 detik
  - Pak Nardi keluar = 2 detik
- total waktu 22 detik

Menaiki lift-3 akan membutuhkan total waktu 44 detik, tidak lebih dari dari menaiki lift-2.





## Problem F

### Belatung Agresif

Wowo, seorang peneliti di sebuah institut penelitian terkemuka di Indonesia, sedang meneliti satu jenis belatung kecil. Belatung ini meskipun kecil namun ternyata memiliki sifat agresif yang cukup berbahaya. Saat ini Wowo memiliki  $N$  buah sarang belatung yang tersebar di beberapa tempat di area penelitiannya. Di setiap sarang belatung terdapat tepat satu “ratu” belatung yang masing-masing bisa memproduksi satu ekor belatung setiap  $D_i$  detik. Area penelitian Wowo berupa lahan berbentuk persegi panjang dengan panjang  $P$  unit dan lebar  $L$  unit. Di sekeliling lahan tersebut, Wowo sudah memasang tembok pembatas setinggi  $H$  unit untuk mencegah agar belatung ini tidak lepas atau keluar dari area penelitiannya.

Suatu hari, karena kelalaian Windi (asisten Wowo), semua sarang belatung di area penelitian ini terkena cahaya radiasi yang mengakibatkan semua ratu belatung dan belatung-belatung lainnya yang diproduksi menjadi lebih pintar dan lebih agresif. Semua belatung ini ingin keluar dari area penelitian dengan memanjat tembok pembatas yang ada. Belatung-belatung ini akan memanjat tembok yang ada dengan cara saling memanjat badan belatung lain sehingga membentuk tumpukan belatung yang tingginya lebih tinggi dari tembok (sehingga belatung yang berada di tumpukan paling atas bisa keluar melewati tembok). Seekor belatung disebut berhasil meloloskan diri dari area jika belatung tersebut berada pada tepi tembok (bersisian langsung dengan tembok) dan berhasil mencapai ketinggian melebihi tingginya tembok.

Asumsikan setiap belatung memiliki dimensi  $1 \times 1 \times 1$  unit<sup>3</sup> dan mereka hanya bisa bergerak ke arah utara, selatan, barat atau timur. Untuk mempermudah permasalahan, asumsikan juga jika ada dua atau lebih belatung pada posisi yang sama, maka secara instan (tanpa memerlukan waktu) mereka akan menjadi sebuah tumpukan belatung. Setiap sarang belatung akan mulai memproduksi belatung pada detik ke  $D_i$ .

Sebagai contoh, misalkan ada sarang belatung A pada posisi  $(4, 2)$  dan sarang belatung B pada posisi  $(2, 3)$  dengan luas area penelitian sebesar 5 unit x 4 unit (lebar x panjang) dan tinggi tembok di sekelilingnya adalah 4 unit. Perhatikan bahwa tembok pembatas ini berada di tepi area penelitian sehingga lebar tembok sebenarnya adalah 7 unit dan panjangnya 6 unit (karakter ‘#’ pada gambar di bawah). Sarang A mampu memproduksi seekor belatung setiap 2 detik, sedangkan sarang B mampu memproduksi seekor belatung setiap 3 detik.

#	#	#	#	#	#
#	$p$				#
#			B		#
#					#
#		A			#
#		$q$			#
#	#	#	#	#	#

Belatung-belatung tersebut bisa memanjat tembok di mana saja, dan karena tinggi temboknya adalah 4 unit, maka diperlukan setidaknya 5 belatung untuk meloloskan diri dari area ini. Contohnya, jika



mereka memanjat tembok pada posisi  $p$  di  $(1, 1)$ , maka waktu yang diperlukan agar ada belatung yang bisa keluar dari area ini adalah:

- A memproduksi 1 belatung pada detik 2 dan belatung tersebut tiba di  $(1, 1)$  pada detik 6.
- B memproduksi 1 belatung pada detik 3 dan belatung tersebut tiba di  $(1, 1)$  pada detik 6.
- A memproduksi 1 belatung pada detik 4 dan belatung tersebut tiba di  $(1, 1)$  pada detik 8.
- B memproduksi 1 belatung pada detik 6 dan belatung tersebut tiba di  $(1, 1)$  pada detik 9.
- A memproduksi 1 belatung pada detik 6 dan belatung tersebut tiba di  $(1, 1)$  pada detik 10.

Dengan demikian, jika mereka memanjat tembok pada posisi  $(1, 1)$  maka total waktu yang mereka perlukan adalah 10 detik. Sedangkan jika mereka memilih untuk memanjat tembok pada posisi  $q$  di  $(5, 2)$ , maka total waktu yang mereka perlukan hanya 9 detik. Belatung dari sarang A akan tiba di  $(5, 2)$  pada detik 3, 5, 7, 9 sedangkan belatung dari sarang B akan tiba di  $(5, 2)$  pada detik 7.

Diberikan informasi luas area penelitian, tinggi tembok serta posisi dan periode produksi setiap sarang belatung yang ada, tentukan berapa waktu minimal yang diperlukan agar setidaknya ada satu belatung yang berhasil meloloskan diri dari area tersebut.

### Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 50$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus diawali dengan tiga buah bilangan bulat  $L$ ,  $P$  ( $1 \leq L, P \leq 1000$ ) dan  $H$  ( $1 \leq H \leq 10^6$ ) yang menyatakan lebar dan panjang area penelitian serta tinggi tembok pembatas. Baris berikutnya berisi sebuah bilangan bulat  $N$  ( $1 \leq N \leq 100$ ) yang menyatakan banyaknya sarang belatung yang ada.  $N$  baris berikutnya masing-masing berisi tiga buah bilangan bulat  $R_i$  ( $1 \leq R_i \leq L$ ),  $C_i$  ( $1 \leq C_i \leq P$ ) dan  $D_i$  ( $1 \leq D_i \leq 1000$ ) yang menyatakan posisi baris dan kolom sarang belatung ke- $i$  serta periode produksi belatung dari sarang tersebut. Asumsikan tidak ada dua sarang yang berada pada posisi yang sama.

### 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 sebuah bilangan bulat yang menyatakan waktu tercepat yang diperlukan agar setidaknya ada seekor belatung yang berhasil meloloskan diri dari area penelitian.

Contoh input	Output untuk contoh input
2 5 4 4 2 4 2 2 2 3 3 4 5 4 2 4 2 2 2 3 3	Kasus #1: 9 Kasus #2: 8



# BNPC

The 2012  
Bina Nusantara Programming Contest  
for High School Students



## Problem G

### Siswa Malas

Windarik adalah seorang siswa sekolah dasar yang cukup pandai namun sangat malas jika diminta mengerjakan tugas berhitung. Di suatu pagi yang cerah dan berseri, Windarik diberi tugas khusus yang terdiri dari banyak soal berhitung oleh gurunya.

Setiap soal diberikan dalam bentuk sebuah persamaan matematika sederhana dengan simbol-simbol operatornya dihilangkan. Tugas Windarik adalah melengkapi operator-operator yang hilang tersebut sehingga persamaan yang diberikan menjadi benar. Operator yang bisa digunakan Windarik hanya tambah ('+') dan kurang ('-'). Karena ini hanya tugas untuk anak sekolah dasar, operand yang diberikan pada soal hanya berupa bilangan bulat antara 0 sampai 9.

Contoh. Sebuah persamaan dengan tiga angka pada ruas kiri: (3, 2, 1) dan 4 angka pada ruas kanan (3 1 2 6) memiliki bentuk:

$$3 \ ? \ 2 \ ? \ 1 \ = \ 3 \ ? \ 1 \ ? \ 2 \ ? \ 6$$

Perhatikan bahwa urutan angka yang diberikan tidak boleh diubah. Dari setiap soal yang diberikan, Windarik harus mengganti semua simbol ? menjadi operator + atau - sedemikian sehingga persamaan tersebut menjadi benar. Salah satu solusi yang memungkinkan adalah:

$$3 + 2 + 1 = 3 - 1 - 2 + 6$$

Solusi lain yang juga benar adalah:

$$3 - 2 - 1 = 3 + 1 + 2 - 6$$

Windarik tidak diperbolehkan menggunakan operator *unary* '-' untuk mengubah tanda positif/negatif pada angka yang diberikan. Sehingga solusi seperti ini tidak diperbolehkan:

$$-3 + 2 + 1 = 3 + 1 + 2 - 6$$

Karena 3 yang merupakan angka pertama pada ruas kiri berubah tanda menjadi -3.

Bantu Windarik untuk menyelesaikan semua soal yang diberikan. Meskipun masih kecil, Windarik bisa cukup mengesalkan jika ia menganggap anda sebagai musuhnya, jadi sebaiknya kita turuti permintaan dia. Diberikan persamaan matematika sederhana yang dihilangkan operator-operatornya, lengkapi persamaan tersebut dengan operator '+' atau '-' sedemikian sehingga persamaan tersebut benar. Jika ada beberapa solusi yang benar, output solusi yang paling kecil secara leksikografis (urutan kamus). Sebagai informasi, karakter '+' memiliki urutan leksikografis yang lebih kecil daripada '-'.



## Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 20$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari 4 baris yang merepresentasikan sebuah persamaan. Baris pertama berisi sebuah bilangan bulat  $N$  ( $1 \leq N \leq 20$ ) yang menyatakan banyaknya bilangan pada ruas kiri. Baris kedua berisi  $N$  buah bilangan bulat  $A_i$  ( $0 \leq A_i \leq 9$ ) yang menyatakan bilangan-bilangan pada ruas kiri secara berurutan. Baris ketiga berisi sebuah bilangan bulat  $M$  ( $1 \leq M \leq 20$ ) yang menyatakan banyaknya bilangan pada ruas kanan persamaan. Baris keempat berisi  $M$  buah bilangan bulat  $B_i$  ( $0 \leq B_i \leq 9$ ) yang menyatakan bilangan-bilangan pada ruas kanan 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 persamaan yang diberikan lengkap dengan operator '+' dan '-' (beserta '=') sedemikian sehingga persamaan tersebut bernilai benar. Output persamaan tersebut tanpa spasi pemisah. Jika ada lebih dari satu persamaan yang memenuhi permintaan, output persamaan yang lebih kecil secara urutan leksikografis (urutan kamus, perlakukan persamaan tersebut secara utuh sebagai sebuah string). Jika tidak ada persamaan yang bisa memenuhi permintaan, output -1 untuk kasus tersebut.

Contoh input	Output untuk contoh input
3 3 3 2 1 4 3 1 2 6 2 5 4 3 2 5 4 2 9 5 1 2	Kasus #1: 3+2+1=3-1-2+6 Kasus #2: 5-4=2-5+4 Kasus #3: -1

### Penjelasan untuk contoh input 1

Persamaan yang memenuhi permintaan pada contoh input 1 adalah:

- $3+2+1=3-1-2+6$
- $3-2-1=3+1+2-6$

Dari dua kemungkinan di atas, pilih kemungkinan yang pertama karena persamaan tersebut lebih kecil secara leksikografis (muncul lebih dulu di dalam kamus).

## Problem H

### Paman Gober

Siapa yang tidak mengenal Paman Gober? Dia adalah bebek terkaya dari kota bebek yang terkenal dengan sifatnya yang sangat pelit. Meskipun demikian, dia adalah seorang pengusaha ulung yang sangat jeli dan pintar dalam memanfaatkan segala kesempatan yang ada untuk meraup keuntungan.

Suatu hari Paman Gober bertemu dengan Nostradabek, seorang peramal hebat yang sedang menjalani pertapaannya di desa angsa. Nostradabek sedang gembira karena dia baru saja dinobatkan sebagai peramal terhebat oleh majalah kota bebek. Paman Gober yang melihat kesempatan ini segera mengunjungi Nostradabek dan meminta bantuan (ramalan tentunya). Paman Gober sangat menginginkan informasi semua pergerakan harga saham yang ada di bursa saham kota bebek selama  $N$  hari ke depan. Tentu saja sebagai peramal yang hebat, Nostradabek bisa memenuhi permintaan Paman Gober.

Untuk mempermudah permasalahan, asumsikan hanya ada 1 jenis saham di bursa saham kota bebek dan harga jualnya selalu sama dengan harga belinya. Saham hanya bisa dibeli dalam jumlah utuh, misalnya jika Paman Gober memiliki 100 rupiah dan harga saham saat itu adalah 30 rupiah per lembar, maka Paman Gober hanya bisa membeli maksimum 3 lembar saham dan menyisakan uang 10 rupiah.

Harga saham selalu berfluktuatif setiap harinya, namun dengan ramalan dari Nostradabek, Paman Gober memiliki informasi harga saham selama  $N$  hari ke depan. Paman Gober bisa membeli atau menjual saham yang ia miliki di hari manapun.

Paman Gober saat ini memiliki dana sebesar  $R$  rupiah. Tugas anda adalah membantu Paman Gober menentukan berapa keuntungan maksimal yang bisa ia dapatkan dan juga menghitung jumlah hari minimal dimana ia harus bertransaksi baik jual ataupun beli agar mendapatkan keuntungan 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 dua buah bilangan bulat  $N$  dan  $R$  ( $2 \leq N \leq 1000$ ;  $1 \leq R \leq 1000$ ) yang menyatakan banyaknya hari di mana Paman Gober memiliki informasi harga saham dan jumlah uang mula-mula yang dimiliki Paman Gober secara berturut-turut. Baris berikutnya berisi  $N$  buah bilangan bulat  $A_i$  ( $1 \leq A_i \leq 100$ ) yang menyatakan harga saham di hari ke- $i$ .

#### Output

Untuk setiap kasus, output dalam satu baris "Kasus #X: A B" (tanpa kutip) dengan X adalah nomor kasus dimulai dari 1 secara berurutan. A adalah total uang maksimal yang bisa dimiliki Paman Gober setelah  $N$  hari, dan B adalah jumlah hari minimum dimana Paman Gober harus bertransaksi (jual ataupun beli) untuk mencapai A. Dijamin tidak ada output yang melebihi  $10^9$ .



# BNPC

The 2012  
Bina Nusantara Programming Contest  
for High School Students



Contoh input	Output untuk contoh input
4 7 10 2 8 3 1 2 5 10 6 25 10 20 30 20 50 10 5 100 80 50 20 10 1 3 50 40 20 90	Kasus #1: 400 4 Kasus #2: 155 4 Kasus #3: 100 0 Kasus #4: 190 2

#### *Penjelasan contoh input 1*

Di hari 1, beli 5 lembar saham seharga 10 (uang = 0, saham = 5).  
Di hari 2, jual 5 lembar saham dengan harga 40 (uang = 40, saham = 0).  
Di hari 4, beli 40 lembar saham seharga 40 (uang = 0, saham = 40).  
Di hari 7, jual 40 lembar saham dengan harga 400 (uang = 400, saham = 0).  
Total dana akhir = 400 dengan 4 kali transaksi.

#### *Penjelasan contoh input 2*

Di hari 1, beli 2 lembar saham seharga 20 (uang = 5, saham = 2).  
Di hari 3, jual 2 lembar saham dengan harga 60 (uang = 65, saham = 0).  
Di hari 4, beli 3 lembar saham seharga 60 (uang = 5, saham = 3).  
Di hari 5, jual 3 lembar saham dengan harga 150 (uang = 155, saham = 0).

#### *Penjelasan contoh input 3*

Melakukan transaksi jual/beli saham tidak akan memberi keuntungan apa-apa.

#### *Penjelasan contoh input 4*

Di hari 2, beli 2 lembar saham seharga 40 (uang = 10, saham = 2).  
Di hari 3, jual 2 lembar saham dengan harga 180 (uang = 190, saham = 0).

## Problem I

### Belajar Berhitung

Natsir baru saja diajari bagaimana cara berhitung dengan bilangan bulat oleh Pak Bowo. Untuk menguji Natsir, Pak Bowo memberikan sebuah barisan bilangan bulat  $A = a_1, a_2, \dots, a_N$ , dan Natsir diminta untuk memilih tepat  $K$  pasang bilangan  $(p_i, q_i)$  yang berbeda dengan  $1 \leq p \leq q \leq N$ . Dua pasang bilangan  $(p, q)$  dan  $(r, s)$  dinyatakan berbeda jika dan hanya jika  $p \neq q$  atau  $r \neq s$ . Dari setiap pasang bilangan  $(p_i, q_i)$  yang dipilih Natsir, ia harus menjumlahkan semua bilangan yang berada di urutan ke  $p_i$  dan  $q_i$  pada  $A$ , yaitu  $a_{p_i} + a_{p_i+1} + \dots + a_{q_i}$  dan mencatatnya sebagai  $S_i$ , kemudian ia harus menjumlahkan semua  $S_j$  dengan  $j = 1..K$  untuk mendapatkan  $R$ , yaitu jumlah dari jumlah semua bilangan yang berada pada rentang  $p_i$  dan  $q_i$  dengan  $j = 1..K$ .

Untuk menambah kesulitan tantangan ini, Pak Bowo menginginkan agar Natsir mencari nilai  $R$  terbesar yang mungkin bisa didapatkan dari  $A$  dan  $K$ .

Contoh, jika  $A = 2, -1, -2, 7, 4$  dan  $K = 3$ , maka kita bisa memilih 3 pasang bilangan  $(p_i, q_i)$ :

- $(1, 5)$ .  $S_1 = a_1 + a_2 + a_3 + a_4 + a_5 = 2 + (-1) + (-2) + 7 + 4 = 10$
- $(2, 5)$ .  $S_2 = a_2 + a_3 + a_4 + a_5 = (-1) + (-2) + 7 + 4 = 8$
- $(1, 4)$ .  $S_3 = a_1 + a_2 + a_3 + a_4 = 2 + (-1) + (-2) + 7 = 6$

Sehingga jumlah  $R$  yang didapatkan dengan 3 pasang bilangan di atas adalah  $10 + 8 + 6 = 24$ . Tentunya  $R$  masih bisa lebih baik lagi jika 3 pasang bilangan yang dipilih adalah:

- $(4, 5)$ .  $S_1 = a_4 + a_5 = 7 + 4 = 11$
- $(1, 5)$ .  $S_2 = a_1 + a_2 + a_3 + a_4 + a_5 = 2 + (-1) + (-2) + 7 + 4 = 10$
- $(3, 5)$ .  $S_3 = a_2 + a_3 + a_4 + a_5 = (-2) + 7 + 4 = 9$

Sehingga total  $R$  yang didapatkan adalah  $11 + 10 + 9 = 30$ , dan ini adalah total tertinggi yang bisa didapatkan dari data  $A$  dan  $K$  di atas.

Bantu Natsir untuk mencari nilai  $R$  terbesar yang bisa didapatkan dari  $A$  dan  $K$ .

#### Input

Baris pertama dari input adalah sebuah bilangan bulat  $T$  ( $T \leq 50$ ) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan dua buah bilangan bulat  $N$  ( $1 \leq N \leq 100000$ ) dan  $K$  ( $1 \leq K \leq N(N-1)/2$ ) yang menyatakan ukuran barisan bilangan yang akan diberikan dan jumlah pasangan bilangan  $(p, q)$  yang harus dipilih. Baris berikutnya berisi  $N$  buah bilangan bulat  $A_i$  ( $-1000 \leq A_i \leq 1000$ ) yang merepresentasikan barisan bilangan yang diberikan.

#### 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  $R$  terbesar yang bisa didapatkan dari  $A$  dan  $K$  pada kasus yang bersangkutan.

*Untuk pengguna C/C++, gunakan `%I64d` atau `cout` untuk mencetak bilangan dengan tipe data integer 64-bit (long long).*





# BNPC

The 2012  
Bina Nusantara Programming Contest  
for High School Students



---

Contoh input	Output untuk contoh input
1 5 3 2 -1 -2 7 4	Kasus #1: 30