

Problem A

Raja yang Bijak

Wacat adalah seorang pangeran yang baru saja diangkat menjadi raja menggantikan ayahnya, Hubu, seorang raja yang terkenal bijaksana. Hubu mampu mengambil segala keputusan yang dapat menyenangkan rakyatnya sekaligus menjaga keuangan kerajaannya dengan baik.

Wacat juga ingin menjadi seperti ayahnya, namun beliau masih perlu belajar lebih banyak mengenai cara mengatur dan mengambil keputusan. Setiap permasalahan i memiliki dua jenis keputusan, baik dan jahat. Setiap keputusan baik akan membuat rakyatnya senang tapi akan mengurangi keuangan kerajaan sebesar A_i (ct: membuka klinik pengobatan gratis), sedangkan keputusan yang jahat tidak akan membuat rakyatnya senang tapi akan menambah keuangan sebesar B_i (contoh: memerintahkan kerja paksa).

Saat ini kerajaan tersebut memiliki harta sebesar S dan Wacat ingin setelah tahun ini berlalu hartanya tidak kurang dari E . Anda akan diberikan N buah permasalahan yang dihadapi Wacat dalam satu tahun dengan setiap permasalahan memiliki dua pilihan keputusan (baik dan jahat), bantu Wacat untuk menentukan berapa banyak keputusan baik maksimal yang bisa ia ambil sedemikian sehingga harta akhirnya tidak kurang dari E , dan output juga berapa harta akhir maksimal yang bisa beliau miliki (dengan mengambil keputusan baik sebanyak mungkin). Jika tidak ada cara bagi Wacat untuk mencapai harta akhir minimal sebesar E , maka beliau harus berusaha agar harta akhirnya mendekati E .

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 30$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan tiga buah bilangan bulat N ($1 \leq N \leq 10.000$), S dan E ($-100.000 \leq S, E \leq 100.000$) yang menyatakan banyaknya permasalahan, harta awal, dan harta akhir minimal secara berurutan. N baris berikutnya masing-masing terdiri dari dua buah bilangan bulat A_i dan B_i ($0 \leq A_i, B_i \leq 1.000$) yang menyatakan bahwa keputusan baik pada permasalahan ke- i akan mengurangi harta kerajaan sebesar A_i dan keputusan jahat akan menambah harta kerajaan sebesar B_i .

Output

Untuk setiap kasus, output dua buah bilangan bulat X dan Y dalam satu baris di mana X menyatakan jumlah keputusan baik maksimal yang bisa diambil sedemikian sehingga harta akhir kerajaan tidak kurang dari E , dan Y adalah harta akhir maksimal yang bisa didapatkan dengan mengambil X keputusan baik.



BNPC

The 2011
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
3 3 0 0 5 20 10 12 7 16 3 0 10 5 20 10 12 7 16 5 10 -20 7 7 2 10 8 3 4 6 5 9	2 3 1 26 5 -16



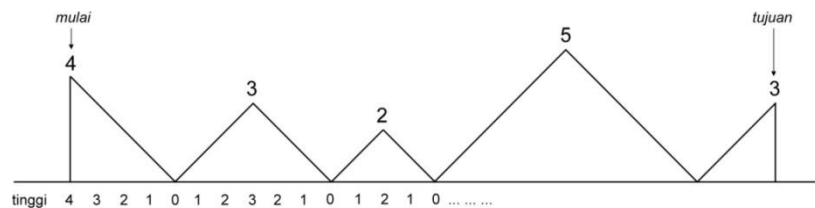
Problem B

Mini Wings

Tiny Wings adalah permainan yang cukup populer di perangkat iPhone. Di permainan ini anda bermain sebagai seekor burung yang berusaha terbang sejauh mungkin melewati bukit-bukit, namun karena sayap burung ini masih kecil, maka ia hanya bisa melaju dengan memanfaatkan momentum ketika sedang merosot menurun bukit. Wacat baru saja menyelesaikan masa kuliahnya di Universitas Bina Nusantara. Melihat kepopuleran permainan Tiny Wings, Wacat tertarik untuk mengembangkan aplikasi serupa yang ia beri nama Mini Wings.

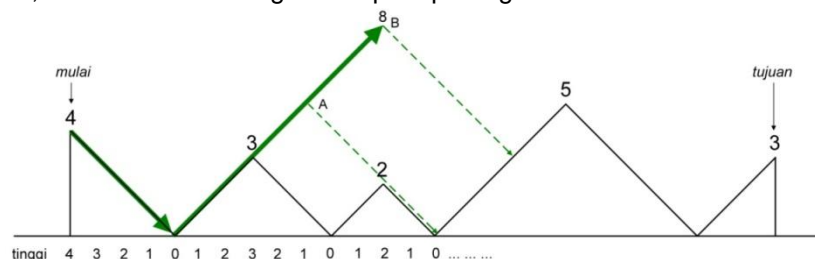
Pada permainan Mini Wings, anda berperan sebagai seekor burung, yang bernama Mini, yang memulai perjalanan dari sisi kiri dan berusaha mencapai sisi kanan dengan melewati bukit-bukit yang ada di antaranya. Tinggi masing-masing bukit akan diberikan dalam format $H_1, H_2, H_3, \dots, H_N$, dengan H_i menyatakan tinggi puncak dari bukit ke i . Setiap bukit memiliki kemiringan yang seragam, yaitu 45 derajat, dan tidak ada jarak pemisah antar dua bukit yang bersebelahan. Anda mulai dari puncak bukit ke-1 dan berusaha mencapai puncak bukit ke- N .

Contoh, $H = \{4, 3, 2, 5, 3\}$, maka struktur bukit pada permainan ini akan tampak seperti gambar di bawah.



Mini akan memulai perjalanan dari puncak bukit-1 dan merosot menurun bukit-1 hingga mencapai ketinggian 0 dan kemudian memanfaatkan momentum pergerakan tersebut untuk mendaki bukit selanjutnya dan jika momentumnya cukup ia akan melambung ke angkasa. Mini bisa mendaki/melambung hingga ketinggian M kali ketinggian bukit yang sebelumnya ia perosoti. Gerakan melambung, turun ataupun merosot semuanya memiliki kemiringan yang sama, yaitu 45 derajat.

Contoh, jika $M = 2$, maka Mini bisa bergerak seperti pada gambar berikut.



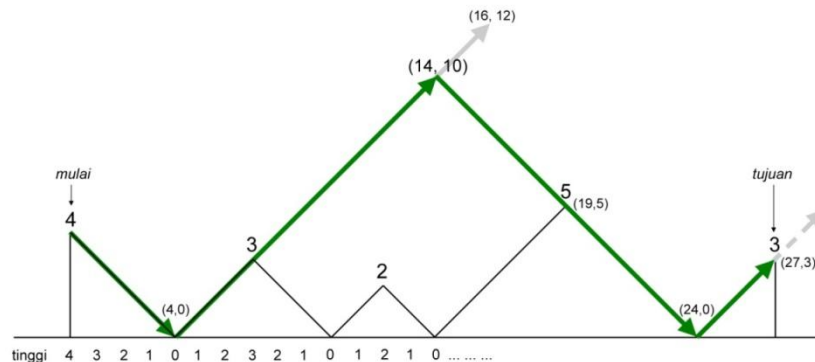
Mini mulai merosot dari puncak bukit-1 yang tingginya 4, ketika mencapai ketinggian 0 ia bisa melambung hingga ketinggian $2 * 4 = 8$. Mini memiliki pilihan untuk berhenti melambung **kapanpun** ia mau. Contohnya, jika ia terus melambung hingga titik B, maka ia akan mulai turun dan akan jatuh di tepi bukit-4 yang tingginya 5 seperti pada gambar. Ia tidak bisa melambung lagi karena ia tidak memerosoti bukit manapun. Jika Mini memutuskan untuk mulai turun pada titik A, maka ia akan memerosoti bukit-3 yang tingginya 2 dan setelah ia mencapai dasar bukit-3 ia akan kembali



melambung hingga ketinggian 4, dan kemudian berhenti (tidak berhasil mencapai titik puncak bukit ke-4). Pada kasus ini tidak ada cara bagi Mini untuk mencapai tujuannya.

Namun jika $M = 3$, maka Mini bisa mencapai tujuannya, salah satunya dengan cara:

- Mulai dari puncak bukit-1, koordinat (0,4), perosoti bukit-1 hingga dasar (4,0).
- Mini bisa melambung hingga (16,12), namun putuskan untuk turun ketika mencapai koordinat (14,10) sehingga Mini bisa mencapai puncak bukit-4 dan memerosotinya.
- Setelah memerosoti bukit-4 Mini bisa melambung hingga (39,15) namun ia hanya perlu mencapai posisi (27,3) yang merupakan tujuannya.



Wacat sangat antusias dengan disain programnya dan sudah merancang beberapa ronde untuk permainan ini. Sekarang Wacat ingin mendisain sistem perolehan nilai dalam setiap ronde, untuk itu Wacat ingin mengetahui pada setiap ronde, berapa kali si Mini harus melambung minimal untuk mencapai tempat tujuannya.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 50$) yang menyatakan banyaknya ronde yang harus ditangani. Setiap ronde dimulai dengan dua buah bilangan bulat dalam satu baris N ($2 \leq N \leq 50.000$) dan M ($1 \leq M \leq 1.000$) yang menyatakan jumlah bukit dan koefisien pelambung secara berurutan. Baris berikutnya terdiri dari N buah bilangan bulat H_i ($1 \leq H_i \leq 1.000.000$) yang menyatakan tinggi bukit ke- i dengan $i = 1..N$ secara berurutan.

Output

Untuk setiap ronde, output dalam satu baris sebuah bilangan bulat yang menyatakan jumlah lambungan minimal yang diperlukan Mini untuk mencapai tujuannya. Output -1 jika tidak ada cara bagi Mini untuk mencapai tujuannya.

Contoh input	Output untuk contoh input
4	2
5 3	-1
4 3 2 5 3	1
5 2	2
4 3 2 5 3	
3 5	
10 30 20	
5 2	
10 2 15 3 7	



Problem C

Bambu Panda

Penduduk desa Panda menggunakan sistem bilangan yang serupa dengan sistem bilangan romawi (I, V, X, L, C, D, M). Namun, mereka tidak memiliki alat tulis, sehingga untuk menuliskan sebuah angka mereka membutuhkan bambu. Jumlah bambu yang diperlukan untuk membentuk sebuah karakter P adalah $A(P)$.

Bambu di desa Panda cukup banyak, tapi tidak gratis, oleh karena itu mereka harus mengetahui berapa banyak bambu yang diperlukan untuk merepresentasikan sebuah angka.

Jika anda tidak cukup mengenal bilangan romawi, baca penjelasan berikut.

Bilangan romawi dibentuk menggunakan 7 jenis simbol:

Simbol	Nilai	Simbol	Nilai
I	1	C	100
V	5	D	500
X	10	M	1000
L	50		

Bilangan romawi dibentuk dengan menggabungkan semua simbol yang tertulis dan menjumlahkan semua nilainya dengan aturan:

1. Simbol I, X, C dan M bisa diulang maksimal 3 kali, sedangkan D, L dan V tidak bisa diulang.
2. I hanya dapat mengurangi V dan X. X hanya dapat mengurangi L dan C. C hanya dapat mengurangi D dan M. V, L dan D tidak bisa digunakan untuk mengurangi.
3. Setiap simbol pengurang hanya boleh digunakan satu kali.
4. Sebuah angka dalam sistem Arabic bisa dipecah menjadi beberapa digit. Contohnya, 1903 terdiri dari digit 1, 9, 0 dan 3. Untuk menuliskan bilangan romawinya, setiap digit non-zero akan diperlakukan secara terpisah. Dalam kasus ini, $1000 = M$, $900 = CM$, $3 = III$, dengan demikian $1903 = MCMIII$.

Diberikan sebuah bilangan N dan jumlah bambu yang diperlukan untuk membentuk masing-masing simbol bilangan Panda, tentukan berapa banyak bambu yang diperlukan untuk membentuk N dalam bilangan Panda.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus berisi 8 buah bilangan bulat N ($1 \leq N \leq 3.000$), $A(I)$, $A(V)$, $A(X)$, $A(L)$, $A(C)$, $A(D)$, $A(M)$ ($1 \leq A(P) \leq 1.000$) yang menyatakan bilangan yang ingin dibentuk dan jumlah bambu untuk membentuk masing-masing simbol I, V, X, L, C, D, M secara berurutan.

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan jumlah bambu yang diperlukan untuk membentuk N.



BNPC

The 2011
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
2 4 1 2 3 4 5 6 7 1000 1 2 3 4 5 6 1000	3 1000



Problem D

Dekat Koq

Pando dan Pancat, dua orang sahabat yang tidak cukup akrab, memutuskan untuk bersama-sama menjenguk Pandut, yang sakit karena terlalu banyak makan, di rumahnya seusai sekolah. Pancat adalah anak yang malas, ia tidak suka berjalan jauh, sedangkan Pando tidak ada masalah dengan jarak yang harus ia tempuh namun ia tidak menyukai persimpangan jalan, lebih tepatnya Pando tidak mau berjalan melewati persimpangan lebih dari K kali.

Diberikan peta yang berisi informasi jalan, lokasi sekolah dan rumah Pandut, bantu Pando dan Pancat untuk menentukan panjang rute terpendek dari sekolah ke rumah Pandut yang tidak melewati lebih dari K persimpangan.

Peta akan diberikan dalam bentuk *grid* berukuran N baris dan M kolom. Setiap kotak/sel berisi salah satu dari karakter:

- 'S' : yang menyatakan sekolah.
- 'P' : yang menyatakan rumah Pandut.
- '#' : yang menyatakan gedung/tembok/penghalang yang tidak bisa dilewati.
- '.' : yang menyatakan jalan setapak yang bisa dilewati.

Pando dan Pancat bisa melewati jalan setapak dan mengunjungi kotak yang bersebelahan dengan kotak yang ia tempati saat itu (utara, selatan, barat, timur). Sebuah kotak disebut persimpangan apabila pada kotak tersebut terdapat lebih dari 1 pilihan untuk ke kotak selanjutnya (tidak termasuk kotak darimana mereka datang).

Untuk lebih jelasnya, perhatikan peta di bawah:

```
1###  
23a.  
#b##
```

1, 2 dan 3 pada peta di atas menyatakan urutan langkah yang dilakukan Pando dan Pancat. Pada posisi 1, kotak tersebut bukanlah merupakan persimpangan (hanya ada satu pilihan jalan), begitu pula dengan posisi 2. Namun pada posisi 3, mereka memiliki dua pilihan jalan, yaitu: a dan b, sehingga kotak pada posisi 3 adalah persimpangan.

Tugas anda adalah untuk mencari panjang rute terpendek dari sekolah untuk mencapai rumah Pandut dengan melewati tidak lebih dari K persimpangan. Jika tidak ada rute yang demikian, output "ooh, tidak isa!" (tanpa tanda kutip).

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus dimulai dengan tiga buah bilangan bulat N ($2 \leq N \leq 100$), M ($1 \leq M \leq 100$) dan K ($0 \leq K \leq 100$) yang menyatakan banyak baris, kolom dan banyak persimpangan maksimal yang boleh dilewati secara berurutan. N baris berikutnya masing-masing terdiri dari M



BNPC

The 2011
Bina Nusantara Programming Contest
for High School Students



karakter yang merepresentasikan peta yang diberikan sesuai dengan yang dijelaskan pada deskripsi permasalahan di atas. Pada setiap peta dipastikan terdapat tepat satu 'S' dan satu 'P'.

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan jumlah langkah minimum yang diperlukan bagi Pando dan Pancat untuk mencapai rumah Pandut dari sekolah tanpa melalui persimpangan lebih dari K kali. Jika tidak ada rute yang demikian, output "ooh, tidak isa!" (tanpa tanda kutip).

Contoh input	Output untuk contoh input
4 4 4 3 S### #. #. ...P 4 4 2 S### #... ...P 6 6 2 S####. #.###.#. #.###.P. 3 4 0 S##P	6 ooh, tidak isa! 11 ooh, tidak isa!

Penjelasan contoh input 3

Jalur yang ditempuh adalah:

```
6 6 2
S####.
123456
#.###7
....#8
#.###9
....10
```

Penjelasan contoh input 4

Pando tidak boleh melewati persimpangan satu kali pun, namun posisi awal sudah mengharuskan mereka untuk memilih jalan ke utara atau ke selatan (sehingga kotak awal adalah persimpangan)



Problem E

Lebah Nakal

Leslie, seekor lebah yang nakal, ingin mencuri madu yang dimiliki seorang bangsawan manusia yang tinggal di hotel yang mewah. Ia mengetahui madu ini karena baunya saja bisa tercium sampai tempat tinggalnya, pastilah ini madu yang enak.

Leslie termasuk lebah yang mengikuti tradisi kuno leluhurnya:

- Pada hari ke- i di mana i adalah kelipatan A dan bukan kelipatan B , lebah harus terbang menaik sejauh i pikometer.
- Pada hari ke- i di mana i adalah kelipatan B dan bukan kelipatan A , lebah harus terbang menurun sejauh i pikometer namun tidak bisa lebih rendah dari ketinggian 0 pikometer.
- Selain itu, lebah harus mempertahankan ketinggiannya.

Tempat tinggal Leslie berada pada ketinggian 0 dan kamar hotel tempat madu tersebut berada pada ketinggian H pikometer. Hari dimana Leslie mulai terbang adalah hari-1. Leslie ingin mengetahui pada hari ke berapa ia bisa mencapai atau melebihi ketinggian H .

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari tiga buah bilangan bulat A , B ($1 \leq A, B \leq 1.000$) dan H ($1 \leq H \leq 10^{18}$) yang menyatakan A dan B sesuai dengan yang disampaikan pada deskripsi permasalahan di atas dan tinggi lokasi madu yang ingin dicapai secara berurutan.

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan hari pertama di mana Leslie berada pada ketinggian H atau lebih. Output -1 jika Leslie tidak mungkin bisa mencapai ketinggian tersebut.

Contoh input	Output untuk contoh input
3	12
3 5 16	20
5 3 20	-1
4 4 27	

Penjelasan contoh input 1

Hari	1	2	3	4	5	6	7	8	9	10	11	12
Tinggi	0	0	3	3	0	6	6	6	15	5	5	17
			+3		-5	+6			+9	-10		+12



Problem F

Barisan Bilangan X

Sebuah bilangan positif N merupakan bagian dari barisan bilangan X jika dan hanya jika memenuhi dua syarat berikut:

1. $F(A) \leq F(N)$ untuk semua $A < N$ dan $A \in X$, di mana $F(P)$ adalah faktor prima terbesar dari P , atau dengan kata lain faktor prima terbesar dari N tidak lebih kecil dari faktor prima terbesar dari bilangan lainnya yang lebih kecil dan termasuk dalam barisan bilangan X .
2. N adalah bilangan komposit (dengan kata lain, lebih besar dari 1 dan bukan bilangan prima).

Beberapa barisan bilangan X yang pertama adalah: 4, 6, 9, 10, ...

- 1 bukan bilangan komposit.
- 2 dan 3 adalah bilangan prima (bukan komposit).
- 4 memiliki faktor prima terbesar 2 dan tidak ada bilangan lain yang lebih kecil dari 4 yang ada di barisan bilangan X yang memiliki faktor prima terbesar yang lebih besar dari 2 (tepatnya, tidak ada bilangan yang lebih kecil dari 4 di dalam X), sehingga 4 adalah bagian dari barisan bilangan X .
- 5 adalah bilangan prima.
- 6 memiliki faktor prima terbesar 3 dan tidak ada bilangan lain yang lebih kecil dari 6 yang ada di barisan bilangan X yang memiliki faktor prima terbesar yang lebih besar dari 3, sehingga 6 adalah bagian dari barisan bilangan X .
- 7 adalah bilangan prima.
- 8 memiliki faktor prima terbesar 2 dan ada bilangan lain yang lebih kecil dari 8 yang ada di barisan bilangan X yang memiliki faktor prima terbesar yang lebih besar dari 2, yaitu 6 (faktor prima terbesar 3).

Diberikan dua buah bilangan bulat A dan B , tentukan ada berapa banyak bilangan yang termasuk dalam barisan bilangan X dari rentang A hingga B , inklusif.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari dua buah bilangan bulat A dan B ($1 \leq A \leq B \leq 1.000.000$).

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan banyaknya bilangan bulat yang merupakan bagian dari barisan bilangan X dalam rentang A dan B , inklusif.

Contoh input	Output untuk contoh input
3	4
4 10	0
8 8	2
8 13	



Problem G

Aritmatika dan Geometri

Barisan aritmatika adalah barisan bilangan di mana selisih dari setiap bilangan/suku yang bersebelahan adalah sama. Contoh: 2, 4, 6, 8 (selisih 2); atau 21, 25, 29, 33 (selisih 4). Sedangkan barisan geometri adalah barisan bilangan di mana rasio atau perbandingan dari setiap bilangan/suku yang bersebelahan adalah sama. Contoh: 2, 4, 8, 16 (rasio 2); atau 5, 15, 45, 135 (rasio 3).

Windarik mendapat PR dari guru matematikanya, diberikan 3 suku pertama dari suatu barisan, ia diminta menentukan suku ke 4 dari barisan tersebut. Permasalahannya adalah, guru Windarik tidak menyebutkan apakah barisan yang diberikan adalah barisan aritmatika atau geometri. Untungnya guru Windarik sudah menjamin bahwa selisih ataupun rasio dari barisan yang diberikan pasti berupa bilangan bulat, dan tidak ada kasus dimana barisan tersebut tidak bisa ditentukan apakah barisan tersebut adalah barisan aritmatika atau geometri. Bantu Windarik menyelesaikan permasalahannya.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 100$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari tiga buah bilangan bulat A , B dan C ($1 \leq A < B < C \leq 1.000.000$) yang menyatakan suku pertama, kedua dan ketiga secara berurutan.

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan suku ke-4 dari barisan yang diberikan.

Contoh input	Output untuk contoh input
4	8
2 4 6	16
2 4 8	33
21 25 29	135
5 15 45	



Problem H

Derita Seorang Siswa

Sebagai siswa kelas XII, tugas yang diterima Joko semakin banyak jika dibandingkan dengan ketika ia berada di kelas XI. Tentu saja karena guru-guru di sekolah Joko ingin murid-muridnya belajar lebih banyak dalam mempersiapkan diri menghadapi ujian akhir.

Joko mendapat N buah tugas dari guru fisiknya di mana masing-masing tugas memiliki poin P_i dan *deadline* (tenggat waktu kapan tugas tersebut masih boleh dikumpulkan) D_i . Joko sudah membaca semua tugas dan bisa memperkirakan berapa hari yang diperlukan untuk menyelesaikan masing-masing tugas, yaitu L_i . Jawaban tugas dikumpulkan melalui email dengan tidak melewati tengah malam pada hari *deadline*, sehingga Joko masih bisa mengerjakan tugasnya tepat pada hari *deadline*. Untungnya guru fisika Joko tidak mengharuskan Joko menyelesaikan semua tugas yang diberikan, tapi tentunya Joko tetap ingin mendapatkan poin sebesar-besarnya. Joko adalah anak yang memiliki konsentrasi tinggi, sekiranya ia mulai mengerjakan tugas, ia akan menyelesaikannya sampai tuntas sebelum memulai tugas yang lain.

Contoh, Joko diberikan tiga buah tugas:

- Tugas 1 : poin 20, *deadline* pada hari ke-5 dan lama pengerjaan 3 hari.
- Tugas 2 : poin 30, *deadline* pada hari ke-6 dan lama pengerjaan 4 hari.
- Tugas 3 : poin 10, *deadline* pada hari ke-2 dan lama pengerjaan 1 hari.

Jika Joko ingin mengerjakan semua tugasnya maka ia memerlukan $3+4+1 = 8$ hari, sedangkan tidak ada tugas yang *deadline*-nya 8 hari, sehingga jelas bahwa ia tidak mungkin bisa menyelesaikan semuanya dalam *deadline* yang diberikan. Joko bisa memilih mengerjakan tugas 3 terlebih dahulu, baru kemudian mengerjakan tugas 2 sehingga total poin yang ia dapatkan adalah $10+30 = 40$.

Hari	Tugas yang dikerjakan	Poin	Deadline
1	Tugas 3	+10	
2	Tugas 2		tugas 3 dikumpulkan
3	Tugas 2		
4	Tugas 2		
5	Tugas 2	+30	<i>deadline</i> tugas 1
6			tugas 2 dikumpulkan

Hari dimana Joko diberikan semua tugas tersebut adalah hari pertama (dan ia bisa langsung mengerjakannya di hari itu juga). Bantu Joko untuk menghitung berapa total poin maksimal yang ia bisa dapatkan dari tugas-tugasnya.

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 tugas yang diberikan. N baris berikutnya masing-masing terdiri dari tiga bilangan bulat P_i ($1 \leq P_i \leq 100$), D_i ($1 \leq D_i \leq 1000$) dan L_i ($1 \leq L_i \leq 1000$) yang menyatakan poin yang didapatkan, *deadline*, dan lama pengerjaan tugas ke- i .



BNPC

The 2011
Bina Nusantara Programming Contest
for High School Students



Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan total poin maksimal yang bisa Joko dapatkan dari tugas-tugas yang diberikan.

Contoh input	Output untuk contoh input
3 3 20 5 3 30 6 4 10 2 1 3 30 4 2 50 3 1 20 5 1 2 60 4 2 30 3 3	40 100 60



Problem I

Angka yang Hilang

Anda diberikan N buah bilangan bulat yang mula-mula terurut dari 1 hingga N dalam sebuah array A dengan index $1..N$. Angka-angka ini kemudian diubah urutan secara rekursif $f(1, N)$ dengan aturan:

$f(a, b)$: ubah urutan angka-angka pada index $a..b$, inklusif.

1. Jika $a \geq b$ maka berhenti, jika tidak maka lanjut ke langkah 2.
2. Atur ulang urutan angka pada index (a, b) sedemikian sehingga angka-angka yang berada pada index genap **relatif terhadap a** berada di depan index ganjil relatif terhadap a . Urutan angka di dalam kelompok yang sama (ganjil atau genap) tidak berubah.

Contoh: $A[] = \{4, 1, 3, 2, 5, 7, 9, 8, 6\}$, dengan $a = 3$ dan $b = 8$.

$A[]$	4	1	3	2	5	7	9	8	6
i	1	2	3	4	5	6	7	8	9
p			0	1	0	1	0	1	

i : index dari $A[]$.

p : index genap (0) atau ganjil (1) relatif terhadap a , didapat dari $(i - a)$ modulo 2.

Ada tiga angka yang berada pada posisi genap: $\{3, 5, 9\}$ dan tiga angka di posisi ganjil: $\{2, 7, 8\}$, sehingga pengaturan (a, b) pada data di atas akan menghasilkan: $\{3, 5, 9, 2, 7, 8\}$.

$A[]$	4	1	3	5	9	2	7	8	6
i	1	2	3	4	5	6	7	8	9

3. Kerjakan secara rekursif $f(a', b')$ dan $f(a'', b'')$ dengan:

- $a' = a$ dan $b' = m$
- $a'' = m + 1$ dan $b'' = b$

di mana m adalah $(a + b) / 2$, dengan $/$ adalah pembagian bilangan bulat (dibulatkan ke bawah).

Diberikan N dan K , cari dimana posisi bilangan K pada array A setelah pemanggilan $f(1, N)$.

Input

Baris pertama dari input adalah sebuah bilangan bulat T ($T \leq 250$) yang menyatakan banyaknya kasus yang harus ditangani. Setiap kasus terdiri dari dua buah bilangan bulat N dan K ($1 \leq K \leq N < 2^{31}$) yang menyatakan banyaknya bilangan pada array A dan bilangan yang dicari secara berurutan.

Output

Untuk setiap kasus, output dalam satu baris sebuah bilangan bulat yang menyatakan posisi bilangan K pada array A setelah $f(1, N)$ dikerjakan.



BNPCHS

The 2011
Bina Nusantara Programming Contest
for High School Students



Contoh input	Output untuk contoh input
3 4 3 7 6 33 2	2 6 18

Penjelasan contoh input 2

A[] 1 2 3 4 5 6 7

f(1, 7) 1 3 5 7 2 4 6

 f(1, 4) 1 5 3 7

 f(1, 2) 1 5

 f(1, 1)

 f(2, 2)

 f(3, 4) 3 7

 f(3, 3)

 f(4, 4)

 f(5, 7) 2 6 4

 f(5, 6) 2 6

 f(5, 5)

 f(6, 6)

 f(7, 7)

A[] 1 5 3 7 2 6 4

Bilangan 6 setelah pemanggilan f(1, 7) selesai terletak di index ke 6.