
Maratona de Programação de 2010

Universidade de São Paulo

Caderno de Problemas

Departamento de Ciência da Computação IME-USP
Domingo, 22 de agosto de 2010.

Problema A: Canhões de Anúbis

Arquivo: *anubis*. [c/cpp/java]

Anúbis está preocupado, soldados franceses estão praticando tiro ao alvo com balas de canhão na esfinge novamente! Dessa vez ele vai tentar impedir que outras partes da esfinge tenham o mesmo destino do seu nariz favorito.

Para parar os canhões, ele pode invocar raios, que destroem tudo em que tocam, inclusive as balas de canhões. Mas os deuses egípcios já não tem tantos seguidores, o que limita os seus poderes.

Anúbis só pode invocar os raios em determinados instantes de tempo. Por sorte um desses condiz com um momento em que as balas estão no ar. Anúbis quer aproveitar essa chance para derrubar o maior número possível de balas.

Acontece que para fazer a invocação, Anúbis precisa temporariamente passar para o nosso plano de existência. Assim ele só pode acertar alvos que estejam todos num mesmo plano.

Ajude Anúbis a derrubar o maior número possível de balas de canhão.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira linha de cada instância contém um inteiro N , onde $1 \leq N \leq 50$, indicando o número de balas de canhão. Cada uma das próximas N linhas contém três inteiros cada, x_i , y_i e z_i , onde $-1\,000 \leq x_i, y_i, z_i \leq 1\,000$, indicando as coordenadas da i -ésima bala de canhão no momento em que Anúbis pode invocar seus raios.

Os franceses tem uma mira muito boa, assim eles as vezes atiram balas que vão grudar uma na outra para aumentar o dano. Logo, não se assuste se houverem pontos repetidos na entrada. Eles representam balas diferentes que estão juntas. Nesses casos cada bala é contada separadamente caso Anúbis decida acertá-las.

Saída

Para cada instância imprima uma linha contendo um único inteiro, o maior número de balas de canhão que Anúbis pode derrubar.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
3	3
3	3
0 0 0	3
0 0 0	
0 0 0	
3	
0 0 0	
1 1 1	
2 2 2	
4	
1 0 0	
0 1 0	
0 0 1	
1 1 1	

Problema B: Banco do Faraó

Arquivo: *banco*. [c/cpp/java]

Pouca gente sabe, mas foi no Antigo Egito que surgiram os primeiros bancos, de uma forma muito semelhante ao que conhecemos hoje. O principal banco era do faraó, que decidia, de tempos em tempos, tomar para o Estado o conteúdo de algumas contas. Isso ocorria da seguinte forma. Dado N , o número de correntistas do Banco do Faraó (era esse o nome do banco), cada conta podia ter uma quantia em menés (moeda do Antigo Egito) que podia ser, inclusive, negativa (indicando que a pessoa devia aquela quantia ao banco), ou seja, o estado de cada conta era um inteiro a_i . O objetivo do faraó era fazer diversas consultas nas contas de seus súditos. Dado um intervalo $[A, B]$ (correspondente as contas $a_A, a_{A+1}, \dots, a_{B-1}, a_B$) o faraó desejava encontrar um subintervalo de soma máxima, ou seja, cujo sequestro pelo Estado renderia ao Faraó a maior quantia de dinheiro. Isso era explicado aos correntistas como sendo uma oferenda a Amon-Ahcid, o Deus egípcio do dinheiro. Fazendo regularmente tais oferendas o deus ficava satisfeito e permitia que o sistema econômico funcionasse perfeitamente. Isso durou surpreendentemente mais de 500 anos, até que num desses sequestros os correntistas se rebelaram, tomaram o palácio e mataram o faraó. O banco foi saqueado e o sistema ruiu. Só se ouviu falar de bancos novamente centenas de anos depois.

Sua tarefa é dado um registro de contas e uma série de consultas, determinar para cada consulta um intervalo de soma máxima.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira linha de cada instância contém um inteiro N , indicando o número de contas no Banco do Faraó, onde $1 \leq N \leq 100\,000$. A segunda linha de cada instância contém N inteiros, entre $-10\,000$ até $10\,000$, indicando os saldos nas contas dos correntistas. A terceira linha contém um inteiro Q , onde $1 \leq Q \leq 100\,000$, indicando o número de consultas que serão feitas. Cada uma das Q linhas seguintes contém dois inteiros A e B , onde $1 \leq A, B \leq N$, indicando o intervalo que deve ser consultado.

Saída

Para cada instância seu programa deve produzir Q linhas na saída, sendo uma para cada consulta. Cada uma dessas linhas deve conter dois inteiros: o primeiro representa a soma do intervalo com maior soma, e o segundo, o número de elementos desse intervalo. Caso haja mais de um intervalo com maior soma, imprima o número de elementos naquele com maior número de elementos.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
3	-1 1
3	3 2
-1 -2 -3	6 4
1	14 4
1 1	2 1
8	0 3
1 2 -1 4 9 8 -1 2	
4	
1 3	
1 4	
2 5	
7 8	
3	
0 0 0	
1	
1 3	

Problema C: It-miha

Arquivo: *itmih.a*. [c/cpp/java]

No Egito antigo as construções das pirâmides são cercadas de muitos mistérios. Muitos pesquisadores consideram que a tecnologia necessária para construí-las não estava disponível na época, e suspeitam que os egípcios tiveram ajuda de extraterrestres para fazê-las. Um exemplo de um desses mistérios são os números de “It-miha”. Na província egípcia de It-miha foi encontrada uma pedra em que uma sequência de números estava gravada. Aparentemente os números não tinham qualquer ligação, até que Poincaré, no final do século XIX conjecturou que os números gravados naquela pedra eram os 500 primeiros inteiros livres de divisores quadrados perfeitos. Um quadrado perfeito é um número que possui raiz quadrada inteira, como 1, 4, 9, 16, 25, etc. Dizemos que um número é livre de divisores quadrados perfeitos se não for divisível por um quadrado perfeito maior que 1. Pode parecer simples para nós, hoje, determinar tais números, mas devemos pensar que naquela época, há mais de 3500 anos, mesmo o sistema de numeração utilizado era outro, e tornava qualquer conta muito difícil. Vale lembrar que os números de “It-miha” são muito frequentes nas construções das pirâmides. A base da pirâmide de Quéops, por exemplo é de 210×210 e sua altura 105 metros. Todas as dimensões são números de “It-miha”!!!

Os primeiros dez números de “It-miha” são 1, 2, 3, 5, 6, 7, 10, 11, 13, 14. Sua tarefa neste exercício será dado N determinar o N -ésimo número de “It-miha”.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira (e única) linha de cada instância contém um inteiro N , onde $1 \leq N \leq 20\,000\,000\,000$.

Saída

Para cada instância seu programa deve imprimir uma linha que contém o N -ésimo número livre de divisores quadrados perfeitos.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
5	1
1	2
2	5
4	17
12	609
371	

Problema D: Mercado do Cairo

Arquivo: *mercado*. [c/cpp/java]

A sua equipe já está fazendo planos para a visita ao Egito. Um dos locais que querem conhecer é o famoso mercado do Cairo. Para economizar tempo, vocês decidiram que vão entrar pela porta no canto sudoeste do mercado e sair pela porta no canto nordeste. Além disso, vocês vão caminhar sempre em direção à saída, ou seja, só vão se deslocar para o norte ou para o leste.

Os vendedores egípcios tem uma regra peculiar. Se você comprar algo de um deles, só poderá comprar novamente de um outro vendedor que seja mais velho. A punição por desrespeitar essa regra é perder uma mão. É claro que isso pode prejudicar sua equipe na final do ICPC. Por este motivo, você acha melhor seguir as tradições locais. Como não é nada elegante dar o mesmo tipo de lembrança para todos seus amigos, você decidiu que, além de seguir as regras do mercado, vai comprar no máximo uma lembrança de cada vendedor. Isto lhe ajudará a ter uma boa variedade de presentes.

O mercado é bem organizado. Os vãos onde as barracas podem ser colocadas possuem a mesma altura e largura. Cada vão é identificado por uma coordenada (x, y) que indica a coluna e linha do mercado que ele se encontra. De uma vista aérea é possível perceber que todos os vãos estão organizados como um quadriculado. As barracas do mercado foram montadas apenas em vãos válidos (e respeitam rigorosamente as medidas do vão). Estando em uma barraca é possível ir para as barracas que ficam estritamente ao norte, ao leste e a nordeste.

Sabendo a idade dos vendedores e a posição da barraca onde cada um trabalha, determine o número máximo de itens que você pode comprar.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira linha de cada instância contém um inteiro N ($1 \leq N \leq 100\,000$), indicando o número de vendedores no mercado. Cada uma das próximas N linhas contém dois inteiros cada, x_i e y_i ($1 \leq x_i, y_i \leq 1\,000$), indicando as coordenadas da barraca em que o i -ésimo vendedor trabalha.

Os vendedores estão listados em ordem de idade, do mais novo para o mais velho. Dois ou mais vendedores podem dividir uma mesma barraca. Nesse caso você pode negociar (ou deixar de negociar) com eles em qualquer ordem.

Ir para o norte significa aumentar o valor de y e ir para o leste significa aumentar o valor de x . Todas as barracas se encontram dentro do mercado.

Saída

Para cada instância imprima uma linha contendo um único inteiro, o número máximo de itens que você pode comprar.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
5	1
1	2
1 1	1
2	2
1 1	3
1 2	
2	
2 1	
1 1	
3	
1 1	
1 2	
2 1	
4	
1 1	
1 2	
2 1	
2 2	

Problema E: Nefertiti, a rainha do Egito

Arquivo: *nefertiti*.*[c/cpp/java]*

Nefertiti foi rainha do Egito, esposa de Akhenaton, e é tida como uma das mais belas mulheres da história do mundo. Seu busto, que pode ser visto no *Altes Museum* em Berlim, Alemanha, comprova sua beleza rara. O escultor inclusive não terminou a obra, deixando de incrustar a córnea no olho direito, para evitar a ira dos deuses. A vida familiar da rainha do Egito obrigava-a a cuidar de diversas coisas, inclusive do cardápio da corte. Akhenaton era conhecido por detestar que a comida se repetisse com frequência, e mesmo em intervalos regulares. Ele desejava que os cardápios não apenas fossem diferentes, como fosse praticamente impossível descobrir quando um prato se repetiria. Isso criou um enorme problema para os cozinheiros do rei, que não apenas tinham de elaborar os diferentes pratos como planejar a ordem em que deveriam ser oferecidos. Nefertiti teve, então, uma ideia. Elaborou uma lista de N pratos, que seriam repetidos. Uma exigência dela era que a diferença entre o prato preparado no i -ésimo dia e i fosse, em módulo, menor que um certo k dado. Tal exigência, além de ser por motivos religiosos, em virtude de obrigações dos egípcios a Ra, se devia também ao fato de que os ingredientes do prato eram conseguidos neste intervalo, e também estavam sujeitos a perder a validade para o consumo. Sua tarefa neste programa é determinar, dado um inteiro N (número de diferentes pratos) e um inteiro K , quantos diferentes planejamentos podemos fazer (que são, na verdade, permutações π de $\{1, 2, \dots, n\}$) que satisfazem a restrição abaixo:

$$|\pi(i) - i| \leq K, \text{ para } i = 1, \dots, N.$$

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira (e única) linha de cada instância contém dois inteiros N e K , onde $1 \leq N \leq 100$ e $1 \leq K \leq 6$.

Saída

Para cada instância imprima uma linha contendo o número de planejamentos diferentes.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
4	3
3 1	6
3 2	19708
10 3	573147844013817084101
100 1	

Problema F: Oráculo de Alexandria

Arquivo: *oraculo*. [c/cpp/java]

Todo computólogo que se preza conhece o livro “O guia do mochileiro das galáxias” (*The Hitchhiker’s Guide to the Galaxy*) e sabe qual é a resposta para a pergunta fundamental sobre a vida, o universo e tudo mais¹. Mas, o que poucos sabem, é que a história de Douglas Adams é baseada em uma lenda egípcia, de um oráculo situado na cidade de Eskendereyya (Alexandria). Alexandria hoje é a maior cidade do Egito, com mais de 4 milhões de habitantes. Fica no delta do Nilo, e estende-se por 32km na costa do Mediterrâneo. Na Antiguidade, a cidade fundada em 331 a.C. por Alexandre, o Grande, foi umas das principais cidades do mundo e lá ficava o Farol de Alexandria (uma das 7 maravilhas do mundo antigo), a Biblioteca de Alexandria (a maior do mundo antigo) além de outras obras fantásticas. A lenda diz também que lá ficava o grande oráculo de Alexandria. Os habitantes da cidade entregavam ao oráculo pequenos bilhetes com números anotados, e recebia de volta um número, que seria a resposta a uma pergunta fundamental do universo relacionada aos dois números dados.

No seu tratado de 227 d.C. Cleómenes de Naucratis (que se tornou administrador de Alexandria quando Alexandre partiu para suas conquistas) relata alguns resultados obtidos do oráculo:

- Dados 8 e 1 o oráculo devolvia 5040;
- Dados 10 e 3, devolvia 280;
- Dados 4 e 2, devolvia 8;
- Dados 21 e 19, devolvia 42.

Estudos modernos dão conta que o que o oráculo devolvia nada mais era que uma generalização do fatorial de um número inteiro. Como sabemos,

$$N! = N \times (N - 1) \times \dots \times 1.$$

O oráculo devolvia para os dados N e K o K -fatorial de N , ou seja,

$$N \times (N - K) \times (N - 2K) \times (N - 3K) \times \dots,$$

em que o produto era feito enquanto a diferença é maior ou igual a 1. Podemos representar o K -fatorial de um número por ele seguido por K exclamações:

- $8! = 5040$;
- $10!!! = 280$;
- $4!! = 8$;
- $21!!!!!!!!!!!!!!!!!!!!!! = 42$

Dizem que ao ler sobre a lenda do oráculo de Eskendereyya, Douglas Adams teve sua inspiração para sua obra. Também, no Egito está a inspiração do Restaurante do fim do universo, mas isso é outra história...

Sua tarefa é dado inteiros N e K determinar K -fatorial de N .

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira (e única) linha de cada instância contém um inteiro N seguido de K pontos de exclamação, onde $1 \leq N \leq 100$ e $1 \leq K \leq 20$.

¹Se você não leu a famosa trilogia de 4 livros, que por acaso são 5, deve começar o quanto antes.

Saída

Para cada instância imprima uma linha contendo o K -fatorial de N .

É garantido que nenhuma instância na entrada possui resultado maior que 10^{18} .

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
4	6
3!	280
10!!!	65835
19!!!!	8
4!!	

Problema G: Os joguinhos de Ramsés

Arquivo: *piramedes*. [c/cpp/java]

Ramsés II foi o mais prestigioso dos faraós egípcios. Reinou entre 1279 a.C. e 1213 a.C. Construiu vários templos, inclusive os famosos templos de Núbia. O mais famoso é um esculpido na rocha, em Abu Simpel, perto da segunda catarata do Nilo, onde o próprio faraó é reproduzido. Com Nefertari e outras esposas teve provavelmente mais de 6 filhos, com quem gostava muito de brincar de um jogo que chamava de “pirâmide mais alta”. O jogo consistia do seguinte. As crianças recebiam pequenos paralelepípedos de diferentes dimensões (que podiam ser rotacionados), e deveriam com estes cubos construir a pirâmide mais alta que conseguissem. Para construí-la não podiam colocar um paralelepípedo maior sobre um menor, ou seja, se o bloco A está sobre o bloco B, tanto a largura como a profundidade de A devem ser menores ou iguais que as de B.

Amen-hotep, primogênito de Ramsés, era muito bom no jogo, e muitas vezes conseguia construir pirâmides muito mais altas que o pai conseguia. Então, Ramsés decidiu chamar o grande matemático da corte, Narmer, para encontrar para cada conjunto de paralelepípedos a maior pirâmide possível, isto é, a pirâmide com a maior altura possível.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias.

A primeira linha de cada instância contém um inteiro N , onde $1 \leq N \leq 15$, indicando o número de blocos. Cada uma das N linhas seguintes possui três inteiros X , Y e Z que indicam as medidas do bloco.

Saída

Para cada instância imprima uma linha contendo a altura da maior pirâmide possível.

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
3	150
5	33
10 10 10	110
50 50 50	
40 40 40	
20 20 20	
30 30 30	
2	
20 20 20	
30 33 10	
2	
100 10 10	
100 12 8	

Exemplo

Exemplo de entrada	Saída para o exemplo de entrada
4	((((
4 0	()))(
4 4	-1
6 63	((())()
7 13	