

Universidade de Brasília

Seletiva da Maratona de Programação UnB/CIC 2016 – Classe Star

15 de outubro de 2016

Coordenação:

Prof. Edson Alves da Costa Júnior (UnB/FGA)

Prof. Guilherme Novaes Ramos (UnB/CIC)

A) Sobre a entrada

1. A entrada de seu programa deve ser lida da *entrada padrão*.
2. Quando uma linha da entrada contém vários valores, estes são separados por um único espaço em branco; a entrada não contém nenhum outro espaço em branco.
3. Cada linha, incluindo a última, contém o caractere final-de-linha.
4. Quando não indicada outra forma, o final da entrada coincide com o final do arquivo.

B) Sobre a saída

1. A saída de seu programa deve ser escrita na *saída padrão*.
2. Quando uma linha da saída contém vários valores, estes devem ser separados por um único espaço em branco; a saída não deve conter nenhum outro espaço em branco.
3. Cada linha, incluindo a última, deve conter o caractere final-de-linha.

C) Sobre os problemas

As situações retratadas nos problemas são inteiramente fictícias e não correspondem à realidade. Nada escrito nos enunciados tem a intenção de desrespeitar o leitor. Tudo foi escrito de maneira a se adequar às situações hipotéticas da melhor maneira possível.

A Sequência de Somos

Limite de Tempo: 1s

As sequências de Somos são sequências de números, definidas por relações de recorrência, descobertas pelo matemático Michael Somos. Para $k > 1$, a sequência Somos- k (a_0, a_1, a_2, \dots) é definida por

$$a_n a_{n-k} = a_{n-1} a_{n-k+1} + a_{n-2} a_{n-k+2} + \dots + a_{n-(k-1)/2} a_{n-(k+1)/2}$$

se k é ímpar e

$$a_n a_{n-k} = a_{n-1} a_{n-k+1} + a_{n-2} a_{n-k+2} + \dots + (a_{n-k/2})^2$$

se k é par, com $a_i = 1$ para $i \leq k$.

Para $k = 2$ e $k = 3$, as expressões acima resultam na sequência de uns ($1, 1, 1, \dots$). Para $k = 4$, a expressão acima se torna

$$a_n a_{n-4} = a_{n-1} a_{n-3} + a_{n-2}^2$$

que pode ser escrita na forma de recorrência

$$a_n = \frac{a_{n-1} a_{n-3} + a_{n-2}^2}{a_{n-4}}$$

Para $k = 5$, a relação de recorrência é

$$a_n = \frac{a_{n-1} a_{n-4} + a_{n-2} a_{n-3}}{a_{n-5}}$$

Embora não seja óbvio a partir destas relações, as sequências Somos- k para, $k \leq 7$, contém apenas números inteiros. Para $k \geq 8$ as sequências podem conter frações.

Dados os valores de k e n , determine o termo a_n da sequência Somos- k .

Entrada

A entrada consiste em T ($1 \leq T \leq 200$) casos de testes. Cada caso de teste é representado por uma única linha com os valores de k ($2 \leq k \leq 7$) e n , separados por um espaço e seguidos de uma quebra de linha, onde n é um inteiro não-negativo tal que o termo a_n é menor do que 2^{64} .

Saída

Para cada caso de teste, a saída do programa deverá imprimir a mensagem “Somos- $k(n) = s$ ”, onde s é o termo a_n da sequência Somos- k , seguida de uma quebra de linha.

Exemplos de entradas	Exemplos de saídas
4	Somos-4(8) = 59
4 8	Somos-5(10) = 83
5 10	Somos-6(11) = 421
6 11	Somos-7(9) = 9
7 9	

B Bots de Dominó

Limite de Tempo: 2s

Marcos está codificando seu primeiro jogo eletrônico: uma versão digital do jogo de dominó. As regras do dominó são as seguintes:

1. o jogo é composto de 28 peças retangulares, divididas em duas metades;
2. cada peça contém, em cada metade, um número entre 0 (zero) e 6 (seis);
3. no início da partida as peças são divididas entre os quatro jogadores, ficando cada jogador com sete peças;
4. começa a partida o jogador que possuir a peça com o número seis em ambas metades;
5. o jogo continua no sentido horário, um jogador por turno;
6. a cada novo turno, o tabuleiro contém apenas dois extremos livres (no início, ambos extremos contém o número seis);
7. o jogador pode jogar apenas uma peça que contenha, em uma de suas metades, um número correspondente a um dos extremos livre: ele conectará os números iguais, e a outra metade se tornará o novo extremo livre;
8. o jogador que não possuir nenhuma peça que possa ser jogada deve passar seu turno para o próximo jogador;
9. o jogo termina quando um jogador joga sua última peça ou quando nenhum jogador conseguir mais realizar uma jogada: neste caso, o jogo termina empatado.

Para o modo de um jogador apenas, Marcos resolveu implementar uma inteligência artificial (Bot) simples, de acordo com as seguintes regras:

1. o Bot recebe as setes peças em uma dada sequência;
2. a cada turno, ele procurará, sequencialmente, uma peça que possa ser jogada: ao encontrar uma, a jogará imediatamente;
3. se a peça a ser jogada pode ser encaixada em ambos extremos, o Bot a lançará no extremo que contém o menor valor numérico.

Escreva um programa que implemente o Bot de Marcos e que simule uma partida entre 4 Bots, determinando o vencedor.

Entrada

A entrada consiste em vários casos de teste, onde o número T ($1 \leq T \leq 1.000$) é dado na primeira linha da entrada.

Cada caso de teste é composto por quatro linhas, onde cada linha contém 14 números N_i ($0 \leq N_i \leq 6$), que correspondem às metades das 7 peças da sequência atribuída ao Bot M ($1 \leq M \leq 4$).

Pode-se considerar que os Bots estão posicionados em um círculo de tal forma que o sentido horário corresponde à sequência crescente da numeração dos Bots.

Saída

Para cada caso de testes deve ser impressa a mensagem “Caso # d : Bot M ”, onde d é o número do caso de teste (cuja contagem tem início no número um) e M é o número do Bot vencedor; ou a mensagem “Caso # d : Empate”, caso a partida termine empatada.

Exemplos de entradas	Exemplos de saídas
2	Caso #1: Bot 1
0 0 0 1 0 2 0 3 0 4 0 5 0 6	Caso #2: Empate
1 1 1 2 1 3 1 4 1 5 1 6 2 2	
2 3 2 4 2 5 2 6 3 3 3 4 3 5	
3 6 4 4 4 5 4 6 5 5 5 6 6 6	
5 6 0 4 3 6 2 6 5 5 6 6 0 3	
1 6 4 6 0 5 0 2 0 0 2 4 1 2	
1 3 4 4 2 3 1 5 3 4 2 2 1 1	
4 5 0 6 3 3 0 1 1 4 2 5 3 5	

C Corridas

Limite de Tempo: 1s

Algumas corridas do calendário internacional de automobilismo são medidas em milhas, de modo que o total de voltas a serem percorridas, assim como a duração total da prova, dependem de vários fatores, como a extensão do circuito, condições climáticas, acidentes, etc. Em certos casos é preciso encurtar a duração da prova, enquanto que em outros a prova pode sofrer atrasos de dias!

Dadas duas das três seguintes informações: início da corrida, fim da corrida e duração da corrida, determine a informação omitida.

Entrada

A entrada consiste em uma sequência de, no máximo, 1.000 casos de teste. Cada caso de teste é representado por uma linha com três valores, separados por espaços em branco: início da prova I , fim da prova F e a duração D da prova. Um destes valores será representado por um símbolo de subtração '-' (e este deve ser o valor a ser determinado), enquanto os outros dois estarão no formato $hh:mm:ss$ ($0 \leq hh \leq 23, 0 \leq mm \leq 59, 0 \leq ss \leq 59$).

Pode-se assumir que a duração da corrida será inferior a 24 horas.

Saída

Para cada caso de teste, a saída deve ser uma das três mensagens a seguir, a depender da informação que foi omitida:

1. Início: $hh:mm:ss$
2. Fim: $hh:mm:ss$
3. Duracao: $hh:mm:ss$

Ao final de cada mensagem deve ser impressa uma quebra de linha.

Exemplos de entradas	Exemplos de saídas
12:00:00 - 02:00:00	Fim: 14:00:00
- 03:30:00 05:45:00	Inicio: 21:45:00
01:23:45 06:07:08 -	Duracao: 04:43:23

D Números Romanos

Limite de Tempo: 1s

Os números romanos foram utilizados na Roma antiga, onde combinações de determinadas letras do alfabeto latino representavam quantidades inteiras positivas. Eles eram a base do sistema numérico vigente na Europa até o século XIV, quando começaram a ser substituídos pelos numerais hindu-arábicos. Ainda hoje é possível encontrar números romanos, seja em sequências de filmes e ou em títulos papais.

A equivalência entre o símbolo romano e seu correspondente decimal é: I = 1, V = 5, X = 10, L = 50, C = 100, D = 500, M = 1000.

Os números são formados combinando estes símbolos e somando os valores respectivos, sendo que os valores devem aparecer em ordem decrescente, da esquerda para direita. Por exemplo XVIII = 18 e MMCLXXI = 2171.

A exceção a esta regra é conhecida como notação subtrativa: se um símbolo de menor valor preceder um de maior valor, então seu valor deve ser subtraído, e não adicionado, ao total. Por exemplo, IX = 9 e CM = 900.

Porém um símbolo só pode preceder os próximos dois símbolos que o sucedem na ordem crescente, e se corresponder a uma potência de dez. Resumidamente,

1. I só pode preceder V e X;
2. X só pode preceder L e C;
3. C só pode preceder D e M.

Dados dois números romanos, determine a soma deles.

Entrada

A entrada consiste em T ($1 \leq T \leq 1.000$) casos de testes. A primeira linha da entrada fornece o valor de T .

Cada caso de teste é composto por apenas uma linha, com dois números romanos A e B ($1 \leq A, B \leq 3.998$), separados por um espaço em branco e seguidos de uma quebra de linha.

Saída

Para cada caso de teste deverá ser impressa a mensagem “Caso # t : C ”, onde t é o número do caso de teste (cuja numeração é sequencial e começa em um) e C é o número romano que corresponde à soma $A + B$. Esta mensagem deve ser seguida de uma quebra de linha.

Pode-se considerar que C é menor do que 4000.

Exemplos de entradas	Exemplos de saídas
4	Caso #1: IV
II II	Caso #2: LVI
IX XLVII	Caso #3: M
CMLXXIV XXVI	Caso #4: MMMCMXCIX
MMD MCDXCIX	

E Matrículas e Lista de Espera

Limite de Tempo: 2s

O sistema acadêmico de uma universidade deve matricular os alunos em disciplinas no início de cada semestre. O registro de cada aluno contém quatro campos: nome, matrícula, posição no fluxo e situação (indica se o aluno está ou não em condição).

As disciplinas tem vagas limitadas, de modo que os candidatos a uma dada disciplina são ordenados segundo os seguintes critérios:

1. os alunos em condição tem precedência em relação aos alunos regulares;
2. se dois alunos estão na mesma situação, tem precedência aquele que tem maior posição no fluxo;
3. se dois alunos tem mesma situação e mesma posição no fluxo, será aplicada a ordem de inscrição no sistema (quem solicitou a disciplina antes tem precedência).

Uma vez ordenados os candidatos, eles são matriculados em sequência, do primeiro até o limite de vagas. Os candidatos não matriculados formam a lista de espera, que respeita a ordenação já estabelecida.

Conhecidos o número de vagas em uma dada disciplina e os candidatos, determine o nome do primeiro candidato da lista de espera, se houver.

Entrada

A entrada consiste em T ($1 \leq T \leq 20$) casos de teste. A primeira linha de um caso de teste informa o número V ($1 \leq V \leq 1.000$) de vagas na disciplina, enquanto a segunda linha consiste no número M ($1 \leq M \leq 100.000$) de candidatos à disciplina.

Em seguida, há M linhas com os dados dos candidatos, separados por espaços em branco: nome N (uma string composta por até 30 caracteres alfabéticos), posição no fluxo P ($1 \leq P \leq 20$) e situação S (um único caractere, 'R' para regular e 'C' para condição). A ordem destas linhas corresponde a ordem de inscrição dos alunos.

Saída

Para cada caso de teste deve ser impressa a mensagem “Caso # t : A ”, onde t é o número do caso de teste e A é o nome do primeiro aluno da lista de espera, seguido de uma quebra de linha. Caso não seja formada uma lista de espera, a saída deve ser a mensagem “Caso # t : Todos foram matriculados”, sem aspas, também seguida de uma quebra de linha.

Exemplos de entradas	Exemplos de saídas
4	Caso #1: Maria
2	Caso #2: Tatiane
3	Caso #3: Todos foram matriculados
Maria 2 R	Caso #4: Heitor
Joao 4 C	
Beto 1 C	
3	
5	
Carlos 3 R	
Ana 5 R	
Lucas 1 R	
Saulo 4 R	
Tatiane 2 R	
5	
3	
Yolanda 3 C	
Mauro 8 R	
Bruno 2 C	
7	
10	
Ana 20 C	
Beto 20 C	
Carlos 20 C	
Douglas 20 C	
Erica 20 C	
Fabio 20 C	
Geralda 20 C	
Heitor 20 C	
Ivo 20 C	
Kleber 20 C	

F Familiares Russos

Limite de Tempo: 1s

Na Rússia, os nomes dos cidadãos são formados por três partes: nome, patronímico e família. Por exemplo, Yuri (nome) Constantinovitch (patronímico, filho de Constantin) Romanov (família).

De forma simplificada, o patronímico é formado a partir do nome do pai mais um sufixo, que depende do sexo do indivíduo e da terminação do nome do pai: “ich”, “ovich”, “evich”, para homens e “ovna”, “evna”, “ichna”, para mulheres. Se Ivan e Sonia são filhos de Petr, então eles tem patronímico Petrovich e Petrovna, respectivamente.

Já a família deriva do nome do patriarca, adicionado de um sufixo de forma semelhante ao patronímico: “ev”, “ov”, “in”, para homens e “eva”, “ova”, “ina”, para mulheres. Por exemplo, Petrov significa “clã de Petr”.

Dado o nome de um indivíduo russo e uma lista de cidadãos, identifique quantos familiares (mesma família) e quantos irmãos (mesma família, mesmo pai) deste indivíduo há dentre os listados.

Nota: Existem outros sufixos e exceções, tanto para o patronímico quanto para a família. Para efeitos do problema, considere apenas os sufixos citados.

Entrada

A entrada consiste em uma série de, no máximo, 100 casos de teste.

A primeira linha de um caso de teste contém o nome de indivíduo russo. A segunda linha contém um natural N ($1 \leq N \leq 250$) que indica a quantidade de cidadãos na lista. As próximas N linhas contém os nomes contidos na lista, um por linha. Os nomes são composto por, no máximo, 100 caracteres alfabéticos maiúsculos, minúsculos ou espaços em branco.

Saída

Para cada caso de teste deverá ser impressa a mensagem “Caso # t : P parente(s), I irmão(s)”, onde t é o número do caso de teste e P e I correspondem ao número de parentes e irmãos do indivíduo citado na primeira linha do caso de teste.

Exemplos de entradas	Exemplos de saídas
Igor Ivanovich Dmitriev	Caso #1: 1 parente(s), 0 irmão(s)
3	Caso #2: 3 parente(s), 2 irmão(s)
Tania Yurievna Vladimirova	
Petr Vladimirovich Dmitriev	
Raissa Igorevna Ivanova	
Ivan Petrovich Yuriev	
5	
Catia Yurievna Dmitrieva	
Petr Petrovich Vladimirov	
Katia Petrovna Yurieva	
Feodor Petrovich Yuriev	
Natasha Ivanovna Yurieva	

G Gabaritos

Limite de Tempo: 1s

Um professor de ensino médio aplica, a cada mês, uma prova de múltipla escolha de 20 questões, sendo que cada questão contém 5 alternativas, sendo que apenas uma delas está correta. Ao perceber que boa parte da turma estava fazendo a prova “no chute”, ele os alertou: com 5 alternativas por questão, são $5^{20} = 95.367.431.640.625$ gabaritos distintos, o que dá uma chance irrisória de acertar toda a prova ao acaso.

Contudo, os alunos observaram que o professor utilizava uma distribuição igualitária dos itens corretos, ou seja, em todas as provas exatamente 4 alternativas A, B, C, D e E eram corretas. Este fato reduzia o número de gabaritos distintos para 305.540.235.000, um número 312 vezes menor.

O professor, ao ser informado das descobertas dos alunos, e apesar da chance de acertar toda a prova ao acaso continuar muito pequena, ficou preocupado: ele quer mudar a distribuição dos itens certo entre as alternativas, mas não sabe como a mudança poderá impactar o número de gabaritos distintos.

Escreva um programa que, dada a distribuição dos itens certos entre as alternativas A, B, C, D e E, compute o número de gabaritos distintos que podem ser formados, ajudando o professor em sua escolha.

Entrada

A entrada consiste em T ($1 \leq T \leq 1.000$) casos de teste, onde o valor de T é informado na primeira linha da entrada.

Cada caso de teste é composto por uma única linha, contendo cinco inteiros A, B, C, D, E ($0 \leq A, B, C, D, E \leq 20$), separadas por um espaço em branco, que correspondem ao número de questões corretas para cada alternativa, com $A + B + C + D + E = 20$.

Saída

Para cada caso de teste deverá ser impressa uma linha com a mensagem “Caso # d : N gabarito(s) distinto(s)”, onde d é o número do caso de teste e N é o número de gabaritos distintos que podem ser formados com a distribuição dada.

Exemplos de entradas	Exemplos de saídas
3	Caso #1: 1 gabarito(s) distinto(s)
20 0 0 0 0	Caso #2: 305540235000 gabarito(s) distinto(s)
4 4 4 4 4	Caso #3: 97772875200 gabarito(s) distinto(s)
3 5 6 2 4	

H Números de Thabit

Limite de Tempo: 1s

Um número natural T_k é denominado um número de Thabit se ele pode ser escrito na forma $T_k = 3 \cdot 2^k - 1$, onde k é um número inteiro não-negativo. Os primeiros quatro números de Thabit são 2, 5, 11, 23. Um número natural maior do que 1 é primo se os únicos divisores deste número são 1 e ele próprio.

Dado um número natural N , classifique-o como primo, número de Thabit, primo de Thabit (número de Thabit que também é primo) ou composto e não Thabit (nem primo, nem Thabit).

Entrada

A entrada consiste em uma série de casos de teste. A primeira linha da entrada contém o número T ($1 \leq T \leq 1.000$) de casos de teste. Cada caso de testes é representado por uma única linha, contendo um número inteiro N ($1 \leq N \leq 1.000.000$).

Saída

Para cada caso de testes deverá ser impressa a mensagem “Caso # t : C ”, seguida de uma quebra de linha, onde t é o número do caso de teste (cuja contagem se inicia no número 1) e C é a classificação do número, conforme apresentado no texto, sem acentos.

Exemplos de entradas	Exemplos de saídas
4	Caso #1: primo
7	Caso #2: numero de Thabit
3071	Caso #3: primo de Thabit
2	Caso #4: composto e nao Thabit
6	