



XIII Maratona Santo Scuderi de Programação

12 de setembro de 2020

Caderno de Problemas

Este caderno contém 7 problemas, as páginas estão numeradas de 1 a 9, não contando esta página de rosto.

Informações Gerais:

A) Sobre a entrada

- 1) A entrada de seu programa deve ser lida da entrada padrão.
- 2) A entrada é composta por um único caso de teste, descrito em um número de linhas que depende do problema.
- 3) 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.
- 4) Cada linha, incluindo a última, contém o caractere final de linha.

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.

Realização:



UNIVERSIDADE
CATÓLICA
DE SANTOS

Patrocínio:



Problema A

Média Móvel

Nome do arquivo fonte: *media*. [c | cpp | java | py]

A Média Móvel é um estimador usado frequentemente para suavizar flutuações curtas e destacar tendências a longo prazo em séries temporais. Durante a pandemia de Covid-19 a média móvel dos últimos 7 dias do número de casos e mortes causadas pelo vírus tem sido comumente utilizada.

Para calcular a média móvel dos últimos N valores em uma sequência de M números inteiros S_i ($1 \leq i \leq M$) faz-se:

$$\text{Média Móvel} = \frac{S_M + S_{M-1} + S_{M-2} + \dots + S_{M-N+1}}{N} = \frac{1}{N} \sum_{i=0}^{N-1} S_{M-i}$$

Sua tarefa neste problema é determinar a média móvel dos últimos N valores de uma sequência de M números inteiros.

Entrada

A primeira linha contém dois inteiros M ($2 \leq M \leq 10^3$) e N ($2 \leq N \leq M$), o número de elementos da sequência de inteiros e o número de elementos da média móvel, respectivamente. A linha seguinte contém M inteiros S_i ($0 \leq S_i \leq 10^4$, para $1 \leq i \leq M$).

Saída

Seu programa deve gerar uma única linha contendo um inteiro, representando a média móvel dos últimos N elementos da sequência S arredondada para o inteiro mais próximo.

Exemplos

Entrada 5 3 7 10 8 2 4	Saída 5
Entrada 10 10 9 3 0 12 5 1 8 9 13 4	Saída 6
Entrada 2 2 4 5	Saída 4

Problema B

Tendência

Nome do arquivo fonte: *tendencia*. [c | cpp | java | py]

A maioria dos veículos de comunicação tem usado a variação da média móvel dos últimos 7 dias em relação a 14 dias atrás para indicar uma tendência de alta, estabilidade ou queda no número de infectados e mortos por Covid-19.

Para saber a tendência, é preciso calcular a variação percentual das médias móveis em um intervalo de 14 dias. Por exemplo, a média móvel do dia 14 será comparada com a média móvel do dia 1°.

Se este percentual for de até 15%, é considerado estável. Se for acima de 15% positivos, está em alta. Se for mais de 15% negativos, está em queda.

Escreva um programa que dadas a média móvel de casos de Covid-19 de 14 dias atrás e do dia atual, determine se a variação percentual entre esses valores representa uma tendência de alta, estabilidade ou queda no número de casos.

Entrada

A entrada contém uma única linha com dois inteiros, A e B ($0 \leq A, B \leq 10^4$), representando o número de casos de Covid-19 de 14 dias atrás e o número de casos do dia atual, respectivamente.

Saída

Seu programa deve produzir uma linha contendo "ALTA" se a variação percentual entre A e B for maior que 15%, "ESTAVEL" (sem acento) se a variação percentual entre A e B for de 15% para mais ou para menos, e "QUEDA" se a variação percentual for maior que 15% negativos.

Exemplos

Entrada 890 722	Saída QUEDA
Entrada 1100 1265	Saída ESTAVEL
Entrada 55 75	Saída ALTA

Problema C

Desafio da String

Nome do arquivo fonte: desafio. [c | cpp | java | py]

Thiago gosta muito de desafios, ele está muito animado a desafiá-lo para solucionar um que ele resolveu ontem a noite. Neste desafio, você tem uma string S contendo somente letras minúsculas (de ‘a’ a ‘z’), e uma lista de “padrões de substituição” é definida. Um padrão de substituição é uma string associada a um caractere. Você pode produzir uma nova string pela substituição de cada caractere pelo “padrão de substituição” associado ao caractere, por exemplo, se você tem uma string “aaaaaa” e o padrão de substituição “a => bc”, você obtém a string “bcbcbcbcbcbcbcb”. Se um caractere não tem um “padrão de substituição” associado, então o caractere não é modificado na string, por exemplo, se você tem a string “aabc” e o padrão de substituição “a => bc, b=>a”, mas nenhum padrão de substituição para o caractere “c”, então a string produzida será: “bcbcac”.

Você consegue solucionar o desafio do Thiago?

Entrada

A primeira linha da entrada contém a string S ($1 \leq |S| \leq 100$), onde cada caractere de S é uma letra minúscula entre ‘a’ e ‘z’, e o inteiro N ($1 \leq N \leq 26$), representando a string que sofrerá as substituições e o número de padrões de substituição, respectivamente. As próximas N linhas contém um caractere C e uma string P_C ($1 \leq |P_C| \leq 10$) separados por um espaço, onde P_C representa o “padrão substituição” para o caractere C . É garantido que um caractere tem, no máximo, um “padrão de substituição” associado a ele.

Saída

Seu programa deve imprimir a string obtida após a execução de todos os seus padrões de substituição.

Exemplos

Entrada	Saída
aaaaaaa 1 a bc	bcbcbcbcbcbcbc
Entrada	Saída
a 1 a bc	bc
Entrada	Saída
a 3 a abc b c c ba	ababa

Problema D

Os Números de Ronaldo

Nome do arquivo fonte: *ronaldo*. [c | cpp | java | py]

Ronaldo é obcecado por números, atualmente ele está representando números usando N palitos de madeira. Como Ronaldo é muito inteligente, ele já criou um método para representar os números. No seu método, ele usa a mesma quantidade de palitos para representar o mesmo dígito:

- 7 palitos para representar o dígito 8.
- 6 palitos para representar os dígitos 0, 6, e 9.
- 5 palitos para representar os dígitos 2, 3 e 5.
- 4 palitos para representar os dígitos 4 e 7.
- 2 palitos para representar o dígito 1.

Usando este método, ele pode representar os números sabendo antecipadamente o número de palitos de madeira que ele precisa. Ronaldo sabe que não pode representar todos os números, o número de palitos de madeira que ele possui é limitado, e portanto, ele não pode representar números que exigem mais palitos de madeira que a quantidade que ele possui. Como ele é obcecado por números, está representando os números positivos começando com o número 1, depois o número 2, depois o 3, e assim por diante, até ele alcançar o número K , que é o primeiro número que ele não pode representar usando o método de Ronaldo com a quantidade de palitos de madeira que ele possui. Você pode ajudar Ronaldo a encontrar este número rapidamente?

Entrada

A entrada é composta por um único inteiro N ($1 \leq N \leq 128$), representando o número de palitos de madeira que Ronaldo possui.

Saída

Seu programa deve imprimir um número inteiro denotando o menor inteiro positivo que não pode ser representado pelo método de Ronaldo.

Exemplos

Entrada	Saída
2	2
1	1
7	10
70	10888888888

Problema E

Incêndio em Oregon

Nome do arquivo fonte: *incendio*. [c | cpp | java | py]

Ronald Krump fez fortuna nos Estados Unidos, mas foi surpreendido por um incêndio quando estava em uma de suas mansões no Oregon. Agora ele precisa abandonar sua propriedade em seu helicóptero particular, que tem uma capacidade de carga C .

Ronald Krump possui um inventário dos N bens que ele mantém na mansão. Esse inventário é uma lista contendo o identificador de cada bem, seu valor e seu peso. Como é muito ganancioso, Ronald Krump quer saber qual o valor máximo que ele poderá levar em bens cujo peso não exceda a capacidade de seu helicóptero.

Ajude o Ronald Krump e escreva um programa que dado o inventário de seus bens e a capacidade de carga de seu helicóptero, informe o valor que maximiza os bens que ele deve levar e cujo peso seja menor ou igual à capacidade do helicóptero.

Entrada

A primeira linha da entrada contém dois inteiros N ($1 < N \leq 25$) e C ($1 < C \leq 10^3$), representando, respectivamente, o número de bens do inventário de Ronald Krump e a capacidade de carga de seu helicóptero. As N linhas seguintes possuem dois inteiros, V_i ($1 \leq V_i \leq 10^4$) e P_i ($1 \leq P_i \leq 100$), o valor e o peso do i -ésimo item do inventário, respectivamente.

Saída

Seu programa deve produzir uma linha contendo um inteiro representando o valor que maximiza o valor total dos bens levados por Ronald Krump.

Exemplos

Entrada	Saída
6 100 400 5 250 20 212 18 300 65 40 13 120 25	1022

Entrada	Saída
4 25 100 25 50 10 50 14 150 30	100

Entrada	Saída
4 50 1000 51 2000 52 3000 53 4000 54	0

Problema F

Pandemia

Nome do arquivo fonte: *pandemia*. [c | cpp | java | py]

Em tempos de pandemia de Covid-19, o acesso a locais que geram aglomeração, como cinemas, casas de espetáculo, ginásios de esportes e estádios de futebol, está proibido. Para tentar conscientizar a população de que aglomerações são perigosas, você foi contratado para simular o número de pessoas que seriam infectadas pela Covid-19 se um dos elementos de um grupo de pessoas aglomeradas for exposta ao vírus do Covid-19.

Nesta simulação, as pessoas estão em um espaço retangular onde em cada metro quadrado há um indivíduo. Sabe-se que cada pessoa tem um grau de resistência diferente ao vírus da Covid-19. Já se sabe também, que o vírus da Covid-19 sofre mutações, e que algumas dessas mutações têm um poder de contágio maior.

Para realizar a simulação, serão fornecidos o comprimento e a largura do espaço retangular onde as pessoas estarão aglomeradas, o grau de resistência a Covid-19 de cada pessoa, a posição da pessoa neste espaço que foi exposta ao vírus da Covid-19 e o poder de contágio do vírus.

O espaço retangular será representado por uma matriz, onde cada célula corresponde ao espaço e 1 m² ocupado por uma pessoa.

O grau de resistência de cada pessoa é um valor de 0 a 5, onde 0 significa nenhuma resistência e 5 a maior resistência possível. O poder de contágio do vírus é um valor entre 2 e 5. Se uma pessoa for exposta a um vírus da Covid-19 com poder de contágio maior que a sua resistência ao vírus, ela é infectada, e também infectará todos os seus vizinhos à esquerda, à direita, acima e abaixo que também tiverem resistência menor que o poder de contágio do vírus. A figura abaixo ilustra um espaço com 5 metros de largura e 7 de comprimento, totalmente ocupado por 35 pessoas. Os valores nas células da matriz representam o grau de resistência ao vírus da Covid-19 de cada pessoa.

0	0	1	2	3	1	3
5	1	0	3	0	1	4
2	2	1	4	1	0	2
3	4	1	0	5	3	1
2	3	4	2	3	4	2

Sua tarefa é escrever um programa para simular a contaminação de uma população aglomerada em um espaço retangular, quando uma pessoa desta população é exposta ao vírus da Covid-19.

Entrada

A primeira linha da entrada contém dois inteiros, M e N ($4 \leq M, N \leq 100$), representando a largura e o comprimento do espaço em que as pessoas estão aglomeradas. Nas M linhas seguintes aparecem N números inteiros R_i ($0 \leq R_i \leq 5$, para $1 \leq i \leq N$), onde R_i é a resistência da i -ésima pessoa de cada linha. A última linha da entrada é composta por 3 inteiros, X ($1 \leq X \leq M$), Y ($1 \leq Y \leq N$) e C ($2 \leq C \leq 5$), representando, respectivamente, a linha e a coluna da posição da pessoa que foi exposta ao vírus da Covid-19 e o poder de contágio do vírus.

Saída

Seu programa deve produzir uma única linha na saída indicando o número de pessoas que foram contaminadas por Covid-19 na simulação.

Exemplos

Entrada 5 7 0 0 1 2 3 1 3 5 1 0 3 0 1 4 2 2 1 4 1 0 2 3 4 1 0 5 3 1 2 3 4 2 3 4 2 2 2 4	Saída 28
Entrada 5 7 0 0 1 2 3 1 3 5 1 0 3 0 1 4 2 2 1 4 1 0 2 3 4 1 0 5 3 1 2 3 4 2 3 4 2 1 2 3	Saída 12
Entrada 4 5 0 0 1 2 3 5 1 0 3 0 2 2 1 4 1 3 4 1 0 5 3 4 4	Saída 0

Problema G

A Bolha

Nome do arquivo fonte: *bolha*. [c | cpp | java | py]

No ano 2020, devido à pandemia, os últimos jogos da fase regular e os *playoffs* da NBA, estão sendo realizados em um complexo de quadras de basquete montados na Disney, os atletas, a comissão técnica e a imprensa estão isolados e os jogos estão sendo realizados sem torcida, o que tornou o complexo conhecido como “A Bolha”.

Para a temporada 2020/2021, a NBA está querendo realizar os jogos com um número reduzido de torcedores, que irão ter que entrar na bolha e seguir várias regras.

Existem várias regras para ingressar na bolha, para deslocamento dentro do hotel, para deslocamento fora do hotel em dias sem jogos e com jogos, mas estes não são de interesse para esse problema.

As regras para assistir os jogos são as seguintes:

- i. Os jogos ocorrem de 2 em 2 dias;
- ii. Cada hotel está associado a um único ginásio e não existem hotéis diferentes associados a um mesmo ginásio, ou seja, o ginásio I está associado ao hotel I ;
- iii. Os jogos serão realizados às 8h, 11h, 14h, 17h, 20h e 23h em cada uma das quadras (6 jogos em cada quadra);
- iv. O torcedor do hotel J pode assistir até 6 jogos, porém o primeiro jogo do dia ele tem que assistir no ginásio J ;
- v. O torcedor que não assistir ao primeiro jogo do dia, não poderá assistir mais nenhum outro jogo deste dia;
- vi. O torcedor deve assistir aos jogos na sequência, ou seja, se quiser assistir três sua única opção é assistir os jogos das 8h, 11h e 14h, não podendo pular algum horário de jogo nem começar a assistir aos jogos em outro horário diferente das 8h;
- vii. Ao término de cada jogo, a organização disponibilizará ônibus com destino a alguns outros ginásios ou aos hotéis;
- viii. Ao término do jogo, o torcedor não poderá permanecer no ginásio em que estava, devendo assistir o próximo jogo em algum ginásio K , desde que exista um ônibus que vá do ginásio J para o ginásio K ($J \neq K$) ou ainda o torcedor poderá optar por voltar ao seu hotel;
- ix. Havendo um ônibus que vá do ginásio J para o ginásio K , não necessariamente existirá um ônibus que vá do ginásio K para o ginásio J ;
- x. Para evitar que o torcedor tenha que voltar para o hotel entre um jogo e outro, em todos os ginásios são servidas refeições sempre antes das partidas. Após o término de uma partida, o ginásio é esvaziado e os ônibus partem todos no mesmo horário para o ginásio de destino ou para os hotéis, o que torna impossível um torcedor sair de um ginásio passar por outro e ir para um terceiro ginásio;
- xi. No dia anterior às partidas, são divulgados os jogos com os horários e os ginásios onde serão realizados.

Os admiradores de basquete que adquiriram o pacote para assistir toda a temporada regular da NBA precisam de sua ajuda para saber de quantas maneiras eles podem assistir aos jogos em cada dia.

Entrada

A primeira linha de entrada contém dois inteiros separados por um espaço, N ($1 \leq N < 1000$) e E ($1 \leq E \leq N^2 - N$), indicando o número de ginásios (numerados de 1 a N) e de linhas de ônibus que saem de um ginásio para outro, respectivamente. Seguem-se, então, E linhas, cada uma com dois inteiros separados por espaços, X e Y ($X \neq Y$ e $1 \leq X, Y \leq N$), indicando que existe um ônibus que sai do ginásio X e chega no ginásio Y e uma última linha contendo dois inteiros Z ($1 \leq Z \leq N$) e W ($2 \leq W \leq 6$), indicando o número do hotel em que está hospedado e o número de jogos que pretende assistir naquele dia.

Saída

Seu programa deve produzir uma única linha na saída indicando o número de maneiras distintas que um determinado torcedor pode assistir aos jogos daquele dia.

Exemplos

Entrada	Saída
4 5 1 2 1 3 2 1 3 4 4 3 2 3	2

Entrada	Saída
3 5 1 2 1 3 2 1 2 3 3 2 1 4	5

Entrada	Saída
4 4 1 2 1 4 3 2 3 4 1 3	0