

1. O cinema de Bytelândia possui 144 poltronas, organizadas em 12 fileiras e 12 colunas. Estas fileiras são numeradas de 1 a 12 e as poltronas também são numeradas de 1 a 12, e todas as poltronas são compradas individualmente para o filme em exibição. Desta forma, se uma pessoa quiser sentar na terceira fileira, quarta coluna, ela deverá comprar um ingresso para a poltrona 3-4.

Sua tarefa é, dado as poltronas já compradas para um determinado filme, dizer se um grupo de N pessoas conseguirá sentar juntos numa mesma fileira. Caso seja possível, informe qual a primeira poltrona da sequência que deve ser comprada pelo grupo. Caso não seja possível que estas pessoas se sentem juntas, você deverá verificar se é possível que estas pessoas assistam a esta sessão em poltronas separadas por toda a sala. Caso não seja possível, deve ser exibida uma mensagem informando a impossibilidade.

Entrada:

Uma sequência de N poltronas ocupadas, $0 < N < 145$, para uma mesma sessão. Cada linha representará uma poltrona ocupada no padrão {numero}{hífen}{número} como, por exemplo, 3-4, o que representa que a 3ª fileira, 4ª poltrona está ocupada. Será lido a sequência de entrada até que a poltrona 0-0 seja recebida. Após a sequência de poltronas, será fornecido um número M , $0 < M < 145$, que representará a quantidade de integrantes do grupo que quer comprar os ingressos.

Saída:

Existem 3 possíveis saídas:

1. Se for possível o grupo sentar numa mesma fileira, informe a primeira poltrona desta sequência. Neste caso deve ser exibida a mensagem “O grupo pode se sentar juntos, a começar pela poltrona x-y”, onde x e y são os valores encontrados.
2. Se não for possível que o grupo fique juntos numa mesma fileira, mas podem comprar ingressos separados na sessão, exiba a mensagem “O grupo pode assistir a sessão, mas em lugares separados.”
3. Se não houver lugares suficientes para todos os membros do grupo, você deve exibir a mensagem “O grupo não conseguiu assistir o filme nesta sessão.”

Não se esqueça do ‘\n’ no final das strings e do ponto final!

2. Um pirata escondeu um tesouro em uma ilha do litoral brasileiro, deixando como vestígio apenas pergaminhos com uma frase de Carlos Drummond de Andrade. Estas frases, por mais que façam sentido, não indicam um local certo para recuperar o tesouro. Ao analisar o pergaminho, você observa que há algo estranho, já que algumas palavras são escritas de forma aparentemente aleatória com letras maiúsculas e minúsculas. Como você sabe que o pirata era muito correto com a escrita, você logo observa que esta é uma pista e percebe que as letras maiúsculas podem formar uma mensagem.

Como são muitos pergaminhos para serem analisados, você decide criar um programa que recebe o conteúdo dos pergaminhos e escreve na tela a frase decifrada do pergaminho.

Entrada

Um string de N caracteres, $100 < N < 1024$, com o conteúdo do pergaminho. Não há caracteres especiais além dos 128 caracteres aceitos pelo ASCII.

Saída

Palavra ou frase decifrada do pergaminho. Não há separação entre as palavras, no caso de ser uma frase.

3. Numa competição estudantil de saltos ornamentais, uma quantidade N de árbitros avaliam o desempenho de cada atleta. A nota final do competidor é obtida da média das notas dos árbitros, descartando a nota mais alta e a nota mais baixa. Neste caso, se houverem 5 árbitros, apenas 3 notas serão utilizadas para a composição da média. Se houver empate na nota mais baixa ou na nota mais alta, apenas uma ocorrência desta nota é descartada.

Você está encarregado de criar um programa que calcule a média de um atleta, recebendo o número N de árbitros, seguidos de suas notas. Ao final você deverá exibir a nota final do atleta.

Entrada:

Um número inteiro N, $4 < N < 1024$, indicando a quantidade de árbitros. Em seguida será recebida uma sequência de N notas, uma por linha. A nota é um número real que varia de 0 a 10 e possui uma casa decimal.

Saída

Um número real, com uma casa decimal, que indica a nota final do atleta. Esta nota é calculada segundo a norma da competição.

4. A cifra de César é uma das mais simples técnicas de criptografia conhecidas. Ela se baseia na substituição de uma letra da frase por uma outra letra deslocada no alfabeto. Uma das formas que podemos obtê-la é através de um parâmetro de deslocamento: se a cifra determinar que o deslocamento é de 5 casas, todo 'A' será substituído por 'F', 'B' por 'E', e assim por diante. Observe que 'Z' equivale a 'E', e que tudo o que não for letra é ignorado na criptografia.

Sua tarefa é criar um programa que criptografe uma mensagem utilizando a cifra de César. Para isso você receberá um número inteiro N que indica o deslocamento e uma sentença para ser cifrada.

Entrada:

Um número N, $2 < N < 25$, indicando o deslocamento para a cifra de César. Em seguida, você receberá uma mensagem, que pode conter até 4096 letras, números, espaços e pontuação. Nesta mensagem não há caracteres acentuados ou especiais, fora dos aceitos pelo ASCII.

Saída:

A mensagem criptografada de acordo com a cifra de César definida.

5. No jogo de dardos, o jogador tem 3 chances para lançar o dardo no alvo e acumular a maior quantidade de pontos. O alvo é dividido em 20 raios, que avançam do centro do tabuleiro para fora. Estes raios são divididos em 4 áreas, cada uma delas com um multiplicador do valor de cada raia. Duas destas áreas possuem o multiplicador 1, também conhecido como Single. Uma destas áreas possui o fator multiplicador 2, também conhecido como Double e outra o fator multiplicador 3, também conhecido como Triple. Se você acertar a raia 14, numa área Single, você acumulará 14 pontos. Se acertar esta mesma raia numa área Double você acumulará 28 pontos e, se acertar uma área Treble, você acumulará 42 pontos.

Além destas áreas, há duas outras áreas centrais no tabuleiro, conhecidas como Outer Bullseye e Bullseye. O Bull é o centro do alvo e vale 50 pontos. O Outer Bullseye é a área próxima ao centro do alvo, mas não pertencente ao centro do alvo, e que vale 25 pontos.

Errar o tabuleiro, ou seja, não acertar o tabuleiro ou acertá-lo fora dos limites definidos, não vale pontos.

Sua missão é criar um programa que receba as 3 jogadas e diga a quantidade de pontos que foram obtidos neste jogo.

Entrada:

Você receberá 3 vezes a mesma sequência:

- Raio R acertado (considere que o Outer Bullseye, Bullseye e fora do tabuleiro pertençam ao raio 0, ou seja, se você acertar uma destas áreas você receberá o valor 0). R varia de 0 a 20.
- Um caractere indicando a área acertada. Se acertar uma área Single, você receberá o caractere 'S'; se acertar Double 'D'; se acertar Triple 'T'; se acertar o Outer Bullseye 'O'; se acertar o Bullseye 'B'; e se errar o tabuleiro receberá 'X'.

Saída:

A quantidade de pontos obtidos pelo jogador.