



Maratona de Programação Pitágoras - Aracati

8 de dezembro de 2018

Este caderno contém 7 problemas; as páginas estão numeradas de 1 a 8.

Informações Gerais

Estas condições são válidas para todos os problemas, exceto quando explicitamente dito.

Nome do programa

1. Sua solução deve ser chamada *problema.c*, *problema.cpp* ou *problema.java*; onde *problema* é a letra maiúscula que identifica o problema.

Entrada

1. A entrada de seu programa deve ser lida da entrada padrão.
2. A entrada consiste de um único caso de teste, que é descrito com uma quantidade de linhas definida no problema. Não há outras informações na entrada.
3. Quando uma linha contém diversos valores, eles são separados por *um único espaço*. Não há outros espaços na entrada. Não há linhas vazias.
4. Apenas o alfabeto em Inglês é utilizado. Não há caracteres com til, acento, trema ou outros símbolos (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Todas as linhas, incluindo a última, contém o tradicional caractere de quebra-de-linha.

Saída

1. A saída de seu programa deve ser escrita na saída padrão.
2. O resultado de um caso de teste deve ser escrito com uma quantidade de linhas definida no problema. Não deve haver outras informações na saída.
3. Quando uma linha contém diversos valores, eles devem ser separados por *um único espaço*. Não deve haver outros espaços na entrada. Não deve haver linhas vazias.
4. Apenas o alfabeto em Inglês deve ser utilizado. Não deve haver caracteres com til, acento, trema ou outros símbolos (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Todas as linhas, incluindo a última, devem conter o tradicional caractere de quebra-de-linha.
6. Para resultados de números reais, arredonde para o número racional mais próximo com a quantidade equivalente de dígitos de precisão. O caso de teste é feito de modo que não haja ambiguidade neste tipo de arredondamento.

Problema A - Dezenas

Konfa gosta de números. Alguns dizem que é por causa de seus 10 dedos, outros que é porque ele não aprendeu a contar além de 100, mas o motivo realmente não importa: Konfa gosta é de *dezenas*.

Faça um programa que, dado um número, diga a Konfa qual é o dígito que representa uma dezena.

Entrada

A entrada consiste de um número inteiro $0 < N \leq 10^6$.

Saída

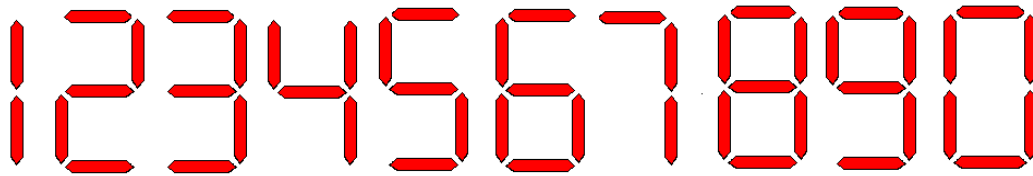
A saída consiste de um número inteiro indicando qual é a dezena.

| | |
|-----------------------------|---------------------|
| Entrada 1 597 | Saída 1 9 |
| Entrada 2 4982919 | Saída 2 1 |
| Entrada 3 7 | Saída 3 0 |

Problema B - LED

Um *diodo emissor de luz* (LED) pode ser usado como uma lâmpada extremamente eficiente e Tio Ernie que utilizá-los para montar o placar em seu novíssimo estádio para Competições de Pinball. Ele sabe que Tommy, atual campeão da competição deve inaugurar o placar, e precisa saber quantos LEDs vai precisar para mostrar a pontuação dos competidores.

Considerando a configuração de LEDs dos números abaixo (cada traço é um LED), faça um algoritmo que ajude Tio Ernie a descobrir o número de LEDs necessário para exibir tal valor.



Entrada

A entrada contém um inteiro N , ($1 \leq N \leq 10^{100}$), correspondendo a pontuação do competidor.

Saída

Mostre uma linha contendo o número de LEDs que Tio Ernie precisará para exibir a pontuação, seguido pela palavra “LEDs” (e quebra de linha!).

| | |
|---|----------------------------|
| Entrada 1 78955645592202548194622645905219891 | Saída 1 177 LEDs |
| Entrada 2 3419174 | Saída 2 26 LEDs |

Problema C - Fizz Buzz

É uma brincadeira de números e palavras. Os jogadores jogam em turnos incrementais, onde cada um diz um número substituindo os números divisíveis por 3 pela palavra “fizz”, e números divisíveis por 5 pela palavra “buzz”. Se o número for múltiplo de 3 e 5, diga “fizzbuzz”.

Por exemplo: “1, 2, fizz, 4, buzz, fizz, 7, 8, fizz, buzz, 11, fizz, 13, 14, fizzbuzz, 16, 17, fizz, 19, buzz, fizz, 22, 23, fizz, buzz, 26, fizz, 28, 29, fizzbuzz, ...”

Faça um programa que leia um número inteiro não negativo e defina o que deve ser dito.

Entrada

A entrada consiste de um número inteiro $0 < n \leq 2^{32} - 1$ que deve ser avaliado.

Saída

A saída deve conter o que deve ser dito conforme a entrada (uma saída por linha).

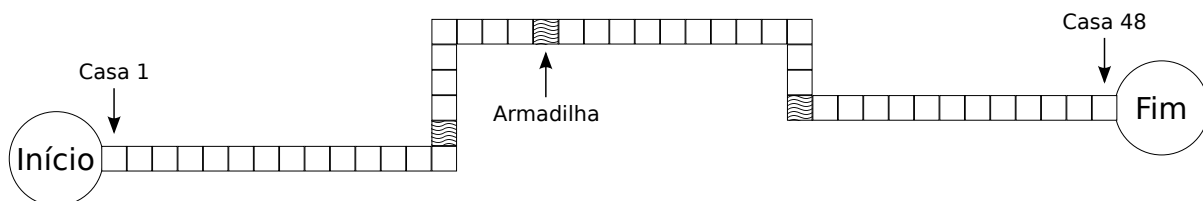
| | |
|--------------------------------|------------------------------|
| Entrada 1 8 | Saída 1 8 |
| Entrada 2 4527 | Saída 2 fizz |
| Entrada 3 2645712641 | Saída 3 2645712641 |

Problema D - Ludo

Um jogo simples jogado por gerações de crianças consiste em um tabuleiro contendo uma trilha de quadrados e um conjunto de peças coloridas. No começo do jogo, cada jogador recebe uma peça e todas as peças são inicialmente posicionadas antes do primeiro quadrado da trilha.

O jogo prossegue em rodadas. Em cada rodada, jogadores lançam um par de dados e avançam suas peças em um número correspondente de casas. A ordem de lançamento dos dados é sempre a mesma (jogador *A*, jogador *B*, etc.), independente do número da rodada.

A maioria das casas no tabuleiro são casas simples, mas três delas são casas de armadilhas. Se a peça de um jogador cai em uma casa de armadilha ao final de sua jogada, sua vez é pulada na próxima rodada. Ou seja, o jogador não lança os dados e sua peça permanece na mesma casa até o fim daquela rodada.



O vencedor é o jogador cuja peça atinge o final da trilha primeiro. O fim da trilha é após a última casa do tabuleiro. Considere, por exemplo, o tabuleiro na figura acima, que possui casas numeradas de 1 a 48. Inicialmente, as peças são posicionadas em uma casa marcada “Início”, antes da casa número 1. Posteriormente, se um jogador lança o valor 7, sua peça será posicionada na casa de número 7 ao final da primeira rodada do jogo. Se a peça futuramente estiver posicionada na casa 41, o jogador precisa lançar um valor de pelo menos 8 para alcançar o fim da trilha e vencer o jogo. Observe também que não há possibilidade de empate.

Neste problema, você receberá o número de jogadores, o número de casas na trilha, a localização das armadilhas e uma lista de lançamento de dados. Você deve escrever um programa que determina quem é o vencedor.

Entrada

A primeira linha de uma entrada contém dois inteiros P e S representando respectivamente o número de jogadores e o número de casas na trilha ($1 \leq P \leq 10$ e $3 \leq S \leq 10000$). A segunda linha descreve as armadilhas, representadas por três inteiros distintos A_1, A_2 e A_3 denotando suas posições na trilha ($1 \leq A_1, A_2, A_3 \leq S$). A terceira linha contém um único inteiro N indicando o número de lançamentos de dados no caso de teste. Cada uma das N linhas seguintes contém dois inteiros D_1 e D_2 ($1 \leq D_1, D_2 \leq 6$) representando os resultados dos lançamentos. O número N em um teste será sempre necessário para um jogador vencer a partida. Cada jogador é identificado por um número entre 1 e P . A ordem de jogadas em uma rodada é de 1 a P .

Saída

Seu programa deve imprimir um único inteiro: o número do jogador vencedor.

| | |
|---|---------------------|
| Entrada 1 2 30 27 11 21 8 5 6 5 6 6 6 4 5 1 1 2 5 3 1 4 1 | Saída 1 1 |
| Entrada 2 3 7 4 5 7 7 1 2 2 2 2 1 1 1 1 2 1 1 1 1 | Saída 2 3 |

Problema E - CD

Você tem uma longa jornada de carro pela frente. Você tem um gravador, mas infelizmente suas melhores músicas estão em CDs. Você precisa tê-las em fitas, então o problema a solucionar é o seguinte: você tem uma fita de N minutos de duração. Como escolher faixas do CD de forma a ter o maior número de faixas na fita, mas com o menor espaço não utilizado possível?

Detalhes:

- O número de faixas no CD não é maior que 20
- Nenhuma faixa é mais longa que N minutos
- Faixas não repetem
- O comprimento de cada faixa é expressado como um número inteiro
- N também é um inteiro

O programa também deve encontrar o conjunto de faixas que melhor preenchem a fita e deve imprimi-los na tela na mesma ordem em que estão no CD.

Entrada

A entrada consiste em 3 linhas, a primeira com o valor $0 < N \leq 1000$, seguido pelo número F de faixas no CD e, por fim, uma linha com a duração de cada uma das faixas do CD.

Saída

Para cada caso de teste, imprima em uma linha o conjunto de faixas (duração em minutos de cada faixa), e em outra linha o texto “total:” seguido da soma das durações.

| | |
|--|------------------------------------|
| Entrada 1 10 2 7 5 | Saída 1 7 total: 7 |
| Entrada 2 5 3 1 3 4 | Saída 2 1 4 total: 5 |
| Entrada 3 10 4 9 8 4 2 | Saída 3 8 2 total: 10 |

Problema F - Mário

Mário é dono de uma empresa de guarda-volumes, a Armários a Custos Moderados (ACM). Mário conquistou sua clientela graças à rapidez no processo de armazenar os volumes. Para isso, ele tem duas técnicas:

- Todos os armários estão dispostos numa fila e são numerados com inteiros positivos a partir de 1. Isso permite a Mário economizar tempo na hora de procurar um armário;
- Todos os armários têm rodinhas, o que lhe dá grande flexibilidade na hora de rearranjar seus armários (naturalmente, quando Mário troca dois armários de posição, ele também troca suas numerações, para que eles continuem numerados sequencialmente a partir de 1).

Para alugar armários para um novo cliente, Mário gosta de utilizar armários contíguos, pois no início da locação um novo cliente em geral faz muitas requisições para acessar o conteúdo armazenado, e o fato de os armários estarem contíguos facilita o acesso para o cliente e para Mário.

Desde que Mário tenha armários livres em quantidade suficiente, ele sempre pode conseguir isso. Por exemplo, se a requisição de um novo cliente necessita de quatro armários, mas apenas os armários de número 1, 3, 5, 6, 8 estiverem disponíveis, Mário pode trocar os armários 5 e 2 e os armários 6 e 4 de posição: assim, ele pode alugar o intervalo de armários de 1 até 4.

No entanto, para minimizar o tempo de atendimento a um novo cliente, Mário quer fazer o menor número de trocas possível para armazenar cada volume. No exemplo acima, ele poderia simplesmente trocar os armários 1 e 4 de posição, e alugar o intervalo de 3 até 6.

Mário está muito ocupado com seus clientes e pediu que você fizesse um programa para determinar o número mínimo de trocas necessário para satisfazer o pedido de locação de um novo cliente.

Entrada

A primeira linha da entrada contém dois números inteiros N e L ($1 \leq N \leq L \leq 10^5$), indicando quantos armários são necessários para acomodar o pedido de locação do novo cliente e quantos armários estão disponíveis, respectivamente. A segunda linha contém L inteiros distintos X_i , em ordem crescente, ($1 \leq X_1 < X_2 < \dots < X_L \leq 10^9$), indicando as posições dos armários disponíveis.

Saída

Seu programa deve produzir uma única linha, contendo um único número inteiro, indicando o número mínimo de trocas que Mário precisa efetuar para satisfazer o pedido do novo cliente (ou seja, ter N armários consecutivos disponíveis).

| | |
|--|---------------------|
| Entrada 1 5 6 1 3 4 5 6 8 | Saída 1 1 |
| Entrada 2 5 5 1 3 5 6 8 | Saída 2 2 |
| Entrada 3 5 6 1 4 5 6 7 8 | Saída 3 0 |

Problema G - Par ou Ímpar

Seu João e Dona Maria viviam discutindo quem deveria lavar a louça, causando um estresse desnecessário ao casamento deles. Felizmente, seu filho propôs uma solução: Seu João lava nos dias pares e Dona Maria nos ímpares.

Faça um programa que, dado o dia, diga quem deve lavar a louça.

Entrada

A entrada consiste de um único inteiro $0 < n \leq 31$, indicando o dia do mês.

Saída

A saída consiste em uma linha indicando quem deve lavar a louça. Imprima exatamente a string “Seu Joao”, caso seja seu João a lavar. Caso contrário, imprima “Dona Maria”.

| | |
|------------------------|------------------------------|
| Entrada 1 3 | Saída 1 Seu Joao |
| Entrada 2 12 | Saída 2 Dona Maria |
| Entrada 3 26 | Saída 3 Dona Maria |