Tutorial: Então é Natal!

Guilherme Novaes Ramos

Há diversas formas de resolver o problema, a mais simples sendo marcar de alguma forma quais foram as renas nomeadas e, ao mostrar todas na ordem da canção, ignorar as marcadas.

Tutorial: Pisca-pisca

Daniel Saad Nogueira Nunes

Esta questão pode ser resolvida da seguinte forma.

Seja g o maior divisor comum entre todos os elementos da entrada. Caso g esteja na entrada, então a classificação do pisca-pisca é g. Caso contrário, o produto está defeituoso.

O valor g pode ser calculado utilizando o algoritmo de Euclides em $\Theta(\lg n)$ para cada par da entrada, o que nos dá um algoritmo com custo total de $\Theta(n \lg n)$.

Tutorial: Papai Noelong

Guilherme Novaes Ramos

Uma abordagem possível é ler o string, converter para um tipo inteiro e acumular os valores, apresentando como resposta o total (convertido em binário).

Outra possibilidade é armazenar os bits em vetores e realizar a soma bit-a-bit.

Tutorial: Papai Noel e os Presentes Errados

Edson Alves da Costa Júnior

Tutorial: Reabastecendo o Trenó

Rodrigo Guimarães Araújo

A distância entre o Papai Noel e o posto i é dada por

$$dist(x_i, y_i) = |x_p - x_i| + |y_p - y_i|$$

Assim, basta computar a distância dele até cada um dos N postos, registrando a distância mínima encontrada. Esta solução tem complexidade $O(N)$.

Tutorial: Cashback

Edson Alves da Costa Júnior

Primeiramente, é preciso observar que se $P > R$, não é possível comprar um vale. No caso em que $P \leq R$, pode-se reservar C dos R reais. Assim, cada novo vale pode ser adquirido por $P - C$ reais. Isto porque, com x reais, pode-se pegar os C reais reservados para completar o valor do vale=presente pois $x + C = (P - C) + C = P$.

Assim, a resposta do problema neste caso é $(R - C)/x$, uma solução tem complexidade $O(1)$.

Tutorial: Dividindo Presentes

Daniel Saad Nogueira Nunes e José Marcos Silva Leite

Em breve

Tutorial: Aposta de Natal

Rodrigo Guimarães Araújo

A probabilidade de que Rafael ganhe a aposta é de $1/N$, e Douglas tem a mesma chance. A probabilidade de ambos vencerem é de $1/N^2$. Assim, a chance de ao menos um deles vencer é de

$$\frac{1}{N} + \frac{1}{N} - \frac{1}{N^2} = \frac{2N-1}{N^2}$$

Esta solução tem complexidade $O(1)$.

Tutorial: Recuperação de Fim de Ano

Edson Alves da Costa Júnior

Primeiramente deve-se observar que a sequência contém apenas frações irredutíveis, isto é, frações p/q com $(p, q) = 1$, por conta do segundo critério. Assim, para um denominador fixo m, haverá $\varphi(m)$ frações com este denominador, onde φ é a função totiente de Euler.

Uma variante do crivo de Erastótenes permite computar os valores de $\varphi(n)$ para $n \in [1, T]$ com complexidade $O(T \log T)$. Em seguida, deve se usar a soma prefixada para acumular estes valores num vetor p_s. Para $T \geq 10^6$ o valor de $p_s(T)$ já ultrapassa o limite de 3×10^{11} apresentado no problema.

Usando busca binária, deve-se localizar qual será o denominador da resposta para, em seguida, usar a busca completa e o maior divisor comum para identificar o numerador. Assim, a solução tem complexidade $O(T \log T)$.

Tutorial: Mega Trenó

Vinicius Ruela Pereira Borges

Para resolver esse problema, primeiramente devemos nos atentar ao fato de que a carretinha só possui uma abertura para que as caixas sejam enfileiradas em seu interior. Isso mostra que a última caixa a ser colocada na carretinha será a primeira a ser retirada, denotando a política de inserção/remoção de dados de uma **pilha**.

Tendo isso em mente, o problema pode ser resolvido fazendo-se uma simulação. Declare duas pilhas P e Q, a primeira associada à carretinha e a outra uma pilha auxiliar para preservamos a ordem de inserção. Utilizaremos também as variáveis $numCidades$ e $jornadas$ para armazenar a quantidade de cidades já visitadas em uma jornada e a quantidade de jornadas realizadas, respectivamente. A variável r armazenará a quantidade de caixas que o Papai Noel já retirou da carretinha (lembrando que ele só consegue retirar, no máximo, D caixas).

Inicialmente, empilhe as M caixas na pilha P. Inicie as variáveis $jornadas \leftarrow 0$ e $numCidades \leftarrow 0$. O algoritmo em pseudo-código que simula o processo de entrega das caixas utilizando a carretinha é descrito a seguir:

enquanto P não for vazia **faça**
 se $numCidades == N$
  $numCidades \leftarrow 0$
  $jornadas \leftarrow jornadas + 1$
 fim-se
 cidade $\leftarrow$ identificador da cidade a ser visitada
 $r \leftarrow D$
 enquanto P não for vazia **E** $r > 0$ **faça**
  $C \leftarrow$ desempilha caixa de P
  se C $\neq$ cidade
   empilha C na pilha Q
  fim-se
  $r \leftarrow r - 1$
 fim-enquanto
 enquanto Q não for vazia **faça**
  $C \leftarrow$ desempilha caixa de Q
  empilha C na pilha P
 fim-enquanto

19. $numCidades \leftarrow numCidades + 1$

20. **fim-enquanto**

A resposta estará na variável $jornadas$.

Tutorial: Seleção de Renas

Daniel Saad Nogueira Nunes

Em breve

Tutorial: Entregando Presentes

Vinicius Ruela Pereira Borges

O problema pode ser resolvido utilizando busca em largura ou profundidade em grafos, pois o grafo é n. Podemos visualizar um grafo com os dados do problema: as N cidades da Nlogonia são associadas aos vértices e as aerovias, que conectam essas cidades, às arestas. A resolução do problema consiste em, iniciando o processo de busca a partir do vértice 1, ir visitando todos os vértices e após o término da busca, deve-se verificar a quantidade de vértices que não foram visitados.

A seguir, apresentamos a solução baseada em **Busca em largura**. Considere uma fila F e um vetor de visitados que serão utilizados para controlar a ordem com que os vértices serão visitados e para marcar os vértices que forem visitados durante a busca, respectivamente. Após construir o grafo a partir dos dados de entrada, pegue o vértice 1, enfileire cada vértice adjacente a 1 e marque-os como visitados. O processo a seguir vai se repetir até que a fila F fique vazia: desenfileire de F um vértice u, enfileire todos os vértices adjacentes a u que não foram visitados, marcando-os como visitados.

Ao final, basta contabilizar a quantidade de vértices que não foram visitados. A complexidade computacional dessa solução é O(N+A).

Tutorial: Manhã de Natal

Rodrigo Guimarães Araújo

Esse problema pode ser simplesmente resolvido lendo cada entrada e comparando com o valor informado e imprimindo o resultado dependendo da existência ou ausência do nome.