

# Aluno: André Santos Rocha

## Ra: 235887

### Problema 1: A - Cubo mágico da Showpee

Neste problema, recebemos  $n \cdot m$  inteiros referentes a um quadrado mágico e outros  $n \cdot m$  inteiros referentes a outro quadrado mágico. Cada inteiro possui uma posição específica no quadrado, como se cada quadrado fosse uma matriz. Em cada quadrado, é permitido realizar operações de troca entre os elementos adjacentes. O objetivo do problema é deixar o primeiro quadrado exatamente igual ao segundo.

#### 1.1 Ideia de solução

Para este problema, podemos utilizar dois vetores Q1 e Q2, um para cada quadrado mágico. Ao receber os elementos do quadrado, inserimo-nos no vetor correspondente ao seu quadrado. Após colocar todos os elementos no quadrado adequado, podemos ordenar os vetores. Perceba que, para que ambos os quadrados possam ser configurados da mesma maneira, eles devem possuir os mesmos elementos nas mesmas quantidades. Logo, após a ordenação, iteramos por um dos vetores, verificando se, para cada elemento  $x$  na posição  $i$  em Q1, o elemento  $y$  na posição  $i$  em Q2 é igual a  $x$ . Caso haja algum  $y$  em Q2 diferente de algum  $x$  em Q1, temos que não é possível igualar os quadrados. Caso contrário, é possível.

#### 1.2 Detalhes de implementação

Sabemos é possível ordenar os vetores com complexidade  $O(n \lg n)$  cada um, de tal forma que é possível passar pelo time limit do problema mesmo com o número máximo  $10^5$  de testes.

### Problema 2: B - quadrados consecutivos

O problema propõe que, dado um inteiro  $n$ , calculemos a diferença  $(n^2 + 1)^2 - n^2$ .

#### 2.1 Ideia de solução

Para este problema, Perceba que  $(n^2 + 1)^2 - n^2 = n^2 + 2n + 1 - n^2 = 2n + 1$ . No entanto, o problema fornece como entrada inteiros  $n$  que são maiores do que os limites de representação numérica das linguagens de programação. Sendo assim, precisamos ler a entrada como um vetor de char  $s$  e performar as operações de multiplicação e soma manualmente, ou seja, da maneira como faríamos no papel. Para multiplicar, para cada dígito, convertamos-no para inteiro e realizamos a multiplicação. Caso o resultado seja

maior ou igual a 10, salvamos o dígito menos significativo em um vetor de char e é preciso salvar um  $\text{carry}=1$ , para adicionarmos no próximo dígito. Caso contrário, o valor de  $\text{carry}$  é 0. Convertemos o resultado para caractere novamente e salvamos a resposta no vetor de char. Após repetir o processo para todos os dígitos, somamos o valor 1 no último dígito. Para isso, convertemos o último dígito para inteiro, realizamos a operação e depois o salvamos como caractere no vetor novamente. Com isso, teremos todas as operações realizadas e basta printar o vetor de char no qual foram salvos os resultados. A variável  $\text{carry}$  é inicializada como 0

## 2.2 Detalhes de implementação

Para converter os valores de char para inteiro, basta realizar a operação  $n = c - '0'$ , onde  $c$  é um número representado em char e  $n$  é o número representado por inteiro. A conversão de inteiro para char pode ser feita pela operação  $c = n + '0'$ .