Soma máxima

A ideia para resolução do problema da soma máxima foi primeiramente obter o número n de elementos inteiros que seriam entrados e armazená-los num vetor de tamanho n. Em segundo lugar, cria-se um outro vetor contendo informações da posição do primeiro elemento e do último elemento analisado, ou seja, o “range” em que a soma máxima se encontrará, além do seu valor.

Para então encontrar esses dados, utiliza-se o uso de aninhamento de loops do tipo for, em que o primeiro representa a posição do primeiro elemento dentro do alcance (“range”), e o segundo a posição do último elemento dentro do alcance analisado. Como em toda iteração do segundo laço há o cálculo da soma dentro do range, caso ela seja maior do que a do range anterior, o vetor de informações é alterado com as informações do primeiro e último elemento do alcance e a respectiva soma.

Análise de complexidade

O algoritmo precisa percorrer em todos os casos os n x n “ranges”, ou alcances, para conseguir determinar a soma máxima, ao utilizar o aninhamento de dois laços do tipo for que vão da posição 1 até a posição n. Então o custo disso é $\Theta(\mathbf{n}^2)$ (teta de n ao quadrado).

Quadrado Mágico

O programa recebe um valor inteiro n e calcula o valor da soma das diagonais, colunas e linhas do quadrado. Após isso, analisa-se por meio de if statements em qual caso n se encaixa, ou seja, se n é ímpar, par não divisível por 4 ou par divisível por 4.

No caso em que n é ímpar, utiliza-se o método Siamês de construção de quadrados mágicos, no qual o primeiro número é colocado em uma posição, e os demais vão sendo colocados em sequência na posição de uma casa acima e para a direita. Caso essa posição não exista, o número é colocado então na última linha e na mesma coluna em que deveria ter sido colocado. Caso já haja algum número na posição, ele é então colocado na posição abaixo.

No caso em que n é par mas não divisível por 4, a estratégia é dividir o quadrado original em quadrados de lado n/2, que serão ímpares, e então aplicar o mesmo algoritmo para resolução de quadrados cujo valor de n é ímpar. Nesse caso em específico, como o programa só suporta n com valor até 6, 6 será o único valor a cair nesse caso. Assim, o range de cada um dos 4 subquadrados deve ser de 1 a 9, 10 a 18, 19 a 27 e 28 a 36. Para resolver isso, deve-se somar à cada um 18, 0, 27 e 9 respectivamente considerando que estivessem dispostos nos 4 quadrantes do plano cartesiano.

No caso em que n é par e divisível por 4, a estratégia utilizada se consiste em criar o quadrado preenchendo-o com cada número em sequência, da esquerda para a direita e de cima para baixo. Após isso, deve-se ordenar as quinas e depois partir para os elementos centrais, sempre computando o valor de $\mathbf{n}^2$ + 1 e subtraindo do valor que a posição antes possuía.

Análise de complexidade

O algoritmo em seu pior caso terá um custo da ordem de $O(\mathbf{n}^2)$ (O grande de n ao quadrado), pois terá que percorrer n x n posições em laços do tipo for aninhados.