

# Trabalho Prático - Matemática Discreta

Quadrado Mágico

E

Soma Máxima

Aluno: Davi Araújo Bady Casseb

Professor: Antonio Alfredo Ferreira Loureiro

**Quadrado Mágico:** Um quadrado mágico de ordem  $n$  é um arranjo de  $n^2$  números, usualmente inteiros distintos, em um quadrado, tais que os  $n$  números em todas as fileiras e colunas somem a mesma constante. Um quadrado mágico contém os inteiros de 1 a  $n^2$ .

A soma constante em cada linha, coluna e diagonal é chamada de constante mágica ou soma mágica,  $M$ . A constante mágica de um quadrado mágico depende apenas de  $n$  e tem o seguinte valor:  $M = n(n^2+1)/2$ .

**Soma Máxima:** para a realização deste exercício foi-se utilizado o algoritmo de Kadane.

-> **Algoritmo de Kadane:** O problema da soma máxima a partir de uma lista de números teve origem numa versão bidimensional mais complexa de um problema de emparelhamento de padrões inicialmente apresentado por **Ulf Grenander**, da **Brown University**.

Ao perceber que o algoritmo cúbico era impraticável para a resolução da versão unidimensional mais simples do problema de identificação de sequências contínuas de maior soma, Grenander desenvolveu a versão quadrática do algoritmo. Posteriormente, Michael Shamos, da atual Carnegie-Mellon University, desenvolveu em 1977 a versão sub-quadrática do algoritmo por divisão e conquista.

Por fim, o estatístico Jay Kadane, da mesma universidade de Shamos, desenvolveu dias depois a versão linear do algoritmo para o problema da sublista contígua de maior soma. Esse algoritmo permanece até os dias atuais como a versão mais eficiente e a melhor solução possível, pois qualquer algoritmo que pretenda resolver o problema em estudo deve necessariamente percorrer os  $N$  elementos da lista dada como entrada.

## Solução linear:

```
void kadane(int *v, int n, int & x, int & y , int & max_total){
    int max_atual;
    int xtemp;
    int i;
    max_atual = 0;
    max_total = -1;
    xtemp = 0;
    for(i=0; i<n; i++){
        max_atual = max_atual + v[i];
        if(max_atual < 0) {
            max_atual = 0;
            xtemp = i+1;
        }
        if(max_atual > max_total){
            max_total = max_atual;
            x = xtemp;
            y = i;
        }
    }
}
```