

Relatório AEDS

Grupo 6:

Felipe Ferraz

Miguel Aspis

Tiago Barros

O problema que foi proposto ao nosso grupo consiste em programar um algoritmo para determinar a existência de um caminho entre todos os vértices de um grafo.

Para resolver esse problema utilizamos um código que gera uma matriz de adjacência, além disso foi realizada uma análise da matriz para verificação dos caminhos entre os vértices.

Após uma busca pelo código é possível verificar que as funções mais complexas são: *cria_grafo* (linha 6), *imprime_grafo* (linha 70), *matriz*(linha 86), *busca_caminhos*(linha 98) e *gera_file*(119) e são de nível $O(n^2)$ pois ele envolve análise de dados em pares, utilizando duas estruturas de repetição aninhadas.

A primeira dificuldade encontrada ao realizar esse trabalho, foi a da representação gráfica de um grafo utilizando *matlab*, onde percebemos que plotar os vértices e posteriormente as arestas era na realidade uma tarefa muito complicada. Para contornar esse problema, utilizamos um círculo, e os vértices são plotados a partir de uma divisão da circunferência pelo número de vértices segundo a fórmula ($X = r * \cos(i * 2\pi / n)$ e $y = r * \sin(i * 2\pi / n)$), que determina as coordenadas dos vértices, e onde i é a iteração e n o número de vértices a partir daí as arestas conectam esses pontos.

Outra dificuldade que nós encontramos, foi na criação e impressão da matriz de adjacência, pois ocorria um nicho de memória em relação ao peso das arestas que conectam os vértices 0 e 1, conseguimos resolver esse problema modificando um ponteiro no momento de alocação dinâmica.