

# **Relatório do Trabalho de Grafos**

**UEM - Universidade Estadual de Maringá**

## **Grafos em Super Mario World**

**Curso:** Ciência da Computação

**Professor:** Felipe Fernandes da Silva

**Alunos:** Gabriel Biscaia e Samuel Amboni

**RA:** 118928 e 100970

# Introdução

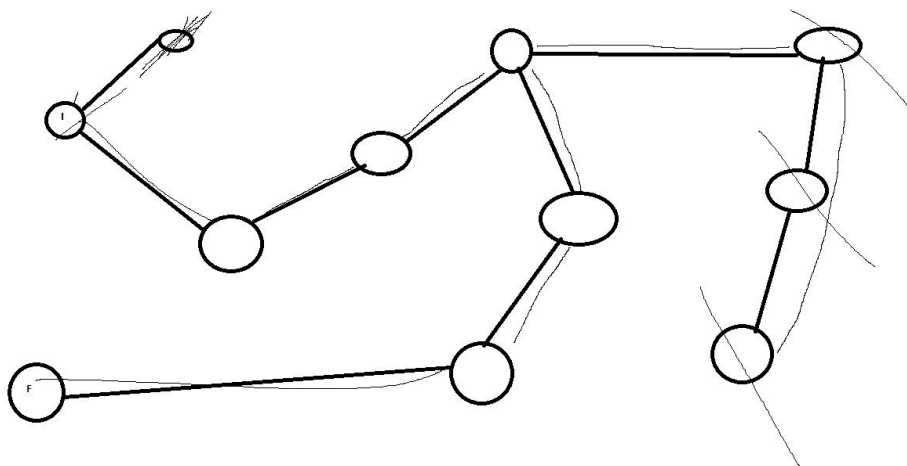
O trabalho em si era dividido em 3 etapas, dentre elas a criação de um leito de arquivo para ler a matriz, o implemento da busca em largura, profundidade e topológica e por fim retornar o menor caminho utilizando Prim ou Kruskal.

## Facilidades e Dificuldades do Trabalho

**Facilidades:** Começando pela primeira etapa (criação da entrada de dados) foi bem tranquila, pois o java oferece ferramentas prontas (bibliotecas) como a “java.io.File” e a “java.util.Scanner” que permitem respectivamente acessar o diretório do arquivo e o lê-lo.

A segunda etapa também foi bem tranquila, pois com a ajuda dos pseudocódigos do professor conseguimos converter a busca em largura e profundidade de maneira rápida, única coisa que a gente teve que refazer foi a busca topológica pois a mesma funciona apenas com grafos orientados, que não é o esse caso.

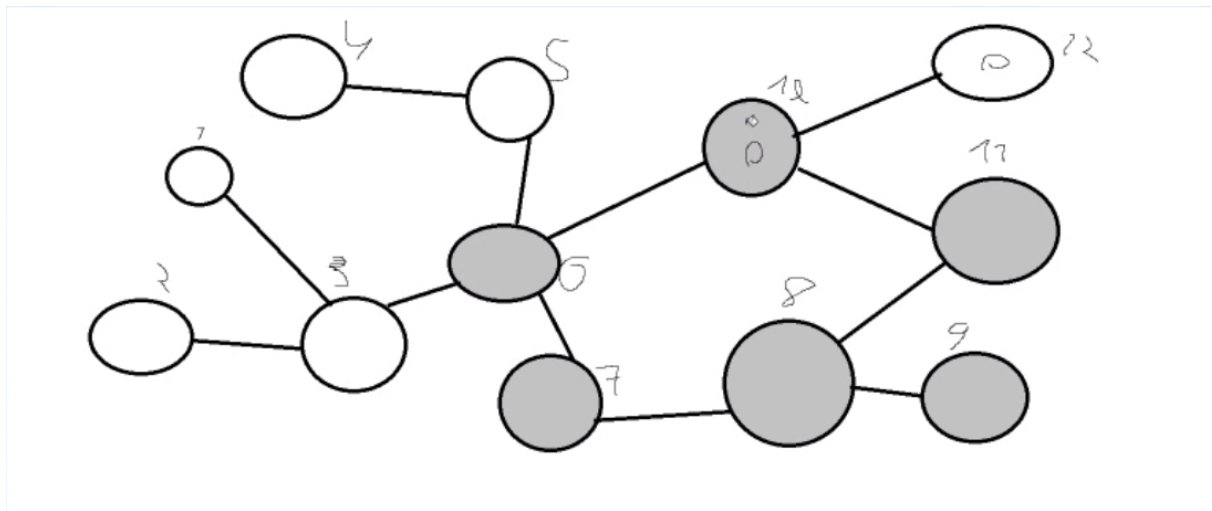
**Dificuldades:** A nossa maior dificuldade no trabalho e também onde nós gastamos mais tempo foi na implementação do prim, no qual a gente usa como referência o vértice zero (yoshi 's house). Nós ficamos muito enrolados nisso, refizemos a função cerca de 5 vezes e nenhuma delas deu certo, até tentamos desenhar a lógica no paint para ver onde estávamos errando, segue a imagem abaixo.



Após isso, tivemos a ideia de criar uma função que teria como intuito apenas retornar o menor caminho, ignorando então os conceitos de *Prim* e *Kruskal* pois ambos precisam dos pesos das arestas. Como este grafo possui apenas arestas de peso 1 chegamos a conclusão que a melhor decisão a ser tomada era focar em apenas retornar o menor caminho possível.

Ademais, no fim do trabalho notamos que tínhamos feito a ordenação topológica de maneira errada, porém isso resolvemos em uma noite, com isso finalizamos o trabalho.

Abaixo está a imagem utilizada como ideia para a nossa ordenação topológica.



## Conclusão

Concluimos que a parte mais difícil do trabalho foi o item 3, o qual chegamos a conclusão que a melhor coisa a ser feita era fazer a nossa lógica para o retorno do menor caminho. Ademais nos enrolamos no topológica porém foi de fácil resolução seu problema.

Os objetivos 1 e 2 do trabalho foram feitos na mesma semana que o trabalho foi dado.

## Observação

Não fizemos a matriz do Mario, usamos matrizes genéricas para testar o funcionamento do mesmo, caso queira o usuário pode inserir no nosso código a matriz do mapa do mario que irá funcionar normalmente.

# Bibliografia

<https://www.todamateria.com.br/como-fazer-um-relatorio/>

<https://www.youtube.com/watch?v=PS44nHjvtdo>

<https://stackoverflow.com/questions/811851/how-do-i-read-input-character-by-character-in-java>

<https://www.delftstack.com/pt/howto/java/how-to-read-a-large-text-file-line-by-line-in-java/#:~:text=o%20m%C3%A9todo%20abaixo.,Scanner%20para%20ler%20arquivo%20linha%20por%20linha%20em%20Java,o%20processa%20linha%20por%20linha.>

**Buscas** = Slides professor felippe