

Relatório - Implementação do Algoritmo de Dijkstra com Visualização Gráfica

Objetivo

Implementar e visualizar o algoritmo de Dijkstra em um grafo utilizando a linguagem Java dentro do ambiente Processing. O grafo é representado com partículas móveis (nós) e arestas com pesos. O objetivo foi destacar graficamente o menor caminho entre dois vértices, utilizando linhas vermelhas para as arestas pertencentes a esse caminho.

Código Implementado

A implementação foi dividida em duas partes principais:

1. Classe Grafo

Responsável por:

- Representar a matriz de adjacência com pesos.
- Calcular a posição e movimentação dos nós (representados como círculos).
- Executar o algoritmo de Dijkstra para encontrar o menor caminho entre dois vértices.
- Destacar as arestas do menor caminho ao desenhar o grafo.

Funções principais da classe:

- `dijkstra(int origem, int destino)`: retorna um `int[]` com os vértices do menor caminho.
- `desenhar(int[] caminho)`: redesenha o grafo, pintando as arestas do caminho em vermelho.

2. Sketch Principal (setup e draw)

Utilizado para:

- Criar o grafo com 6 vértices.
- Adicionar arestas com pesos.
- Chamar o método de Dijkstra.
- Desenhar a simulação interativa.

Relatório - Implementação do Algoritmo de Dijkstra com Visualização Gráfica

Resultados Gráficos

A execução do código no Processing gerou a seguinte visualização:

- Os nós do grafo foram distribuídos circularmente na tela e representados por círculos brancos com números.
- As arestas foram desenhadas como linhas pretas com espessura proporcional ao peso.
- Após a execução do algoritmo de Dijkstra, as arestas que pertencem ao menor caminho entre os vértices de origem (0) e destino (5) foram desenhadas em vermelho e com espessura maior.
- A movimentação dos nós é suavizada por forças físicas de atração e repulsão, dando vida ao grafo.

Dificuldades Encontradas

Durante a implementação, as principais dificuldades foram:

1. Conversão do algoritmo de Dijkstra do pseudocódigo para Java:

- Foi necessário adaptar o algoritmo para trabalhar com matrizes de adjacência com pesos e manipular arrays de distâncias, predecessores e visitados corretamente.

2. Destaque das arestas corretas no caminho:

- Para destacar as arestas corretas no desenho, foi necessário implementar uma função auxiliar (fazParteDoCaminho) que verifica se a aresta conecta vértices consecutivos do array retornado por Dijkstra.

3. Integração com o sistema de visualização física:

- Como os nós são móveis devido às forças físicas simuladas, foi importante manter a atualização das posições sincronizada com o desenho das arestas e do caminho.

Conclusão

A implementação foi bem-sucedida, atendendo aos seguintes objetivos:

Relatório - Implementação do Algoritmo de Dijkstra com Visualização Gráfica

- Execução correta do algoritmo de Dijkstra para encontrar o menor caminho entre dois vértices.
- Representação gráfica clara e funcional do grafo com destaques visuais.
- Separação modular entre lógica de cálculo e visualização.

O sistema permite fácil modificação e extensão para suportar novos algoritmos ou interações com o usuário. Essa experiência contribuiu para o aprendizado de algoritmos em grafos e sua aplicação gráfica com feedback visual imediato.