
Informações:

- i. A linguagem de programação utilizada pode ser a de sua preferência.
 - ii. Os algoritmos implementados devem seguir aqueles com a **menor complexidade possível** em termos assintóticos.
 - iii. Somente serão considerados para avaliação as operações que foram solicitadas explicitamente.
 - iv. A clareza e concisão das implementações também são objetos de avaliação.
 - v. Comentários de ajuda no código também serão levados em consideração.
 - vi. O projeto pode ser implementado em grupos de até 2 pessoas.
 - vii. **O prazo máximo para entrega é o dia 15/05/2025.**
 - viii. **A nota do projeto corresponderá a 25% da média final.**
 - ix. **As notas no projeto serão individuais.**
 - x. A entrega do projeto consistirá de forma online (github, por exemplo) contendo: os arquivos de código; um arquivo com as instruções para a execução do seu código; **links** para vídeos contendo a explicação da implementação, um para cada participante da equipe; no vídeo deverá ser explicado por qual parte você foi responsável e como foram implementadas as soluções desta parte. **Não envie arquivos de vídeo.**
-

Especificações:

Na região do Cariri deseja-se implementar um sistema de prevenção e combate a incêndios em parques florestais e áreas rurais da região. Para isso, deseja-se instalar postos de brigadas de combate a incêndios para atuar em necessidades e de forma mais célere.

Considere que toda a região foi mapeada em termos de rotas de acesso e áreas onde é possível haver ocorrências de incêndios. Assim, uma floresta ou região rural pode ter várias rotas de acesso dentro dela, cada uma com um nível de dificuldade já mapeados e traduzidos em uma função que leva em conta a inclinação do terreno, densidade e tipo da vegetação, presença de residências próximas, etc.

1 Introdução

O combate a incêndios em áreas vegetais é um problema crítico que pode ser modelado utilizando grafos. Neste projeto, deve-se implementar um sistema que represente regiões suscetíveis a queimadas como um grafo, onde:

- **Vértices** representam locais da vegetação.
- **Arestas** indicam conexões entre os locais, com um custo associado ao deslocamento das equipes de brigadistas.
- **Postos de brigadistas** são distribuídos aleatoriamente entre os vértices.
- **Pontos de coleta de água** fornecem recursos para o combate ao fogo.
- **O fogo pode se propagar** para vértices vizinhos ao longo do tempo.
- **A propagação do fogo** se dá em tempos discretos, ou seja, todos os vértices que possuem algum vizinho com fogo serão afetados ao mesmo tempo.
- **Objetivo:** maximizar a quantidade de vértices salvos após o início do fogo.

2 Especificação do Problema

O sistema deverá simular a propagação do fogo e a resposta das equipes de brigadistas. Os elementos do problema são:

2.1 Postos de Brigadistas

- Três equipes alocadas aleatoriamente em vértices distintos no grafo.
- Cada posto possui a mesma quantidade de equipes de brigadistas e caminhões de combate a incêndios.
- Brigadistas podem se mover por toda a região seguindo as rotas definidas pelo grafo.

2.2 Pontos de Coleta de Água

- São localizações onde os caminhões podem reabastecer.
- Estes podem ser lagos, lagoas, açudes, rios, etc.
- **Os postos de brigadistas também servem para reabastecimento de água.**
- Cada **ponto de coleta é representado também por um vértice** conectado aos demais pontos.

2.3 Caminhões de Brigadistas

- Possuem um **limite de armazenamento de água** (parâmetro de entrada).
- Precisam se deslocar até o local do incêndio seguindo um caminho de menor custo.
- Uma vez esvaziados, os caminhões devem reabastecer em algum ponto de coleta mais próximo.
- Considere que o tempo de reabastecimento é irrisório, ou seja, uma vez alcançado um ponto de coleta o caminhão já pode retornar ao combate ao incêndio.

2.4 Propagação do Fogo

- O fogo se inicia em um vértice e pode avançar para vizinhos a cada unidade de tempo.
- Para extinguir um incêndio em um vértice, é necessário um volume específico de água e uma equipe de brigadistas.
- Se o fogo não for contido a tempo, ele se espalha, mas vértices já queimados não propagam novos focos.

2.5 Simulações

- Deseja-se simular o combate ao fogo iniciado em diferentes vértices do grafo e a resposta do sistema será analisada, para cada vértice que não contém um posto de brigadistas.
- O resultado será um relatório com a quantidade de vértices salvos em cada simulação.

3 Entradas e Saídas

3.1 Entradas

- Um grafo $G = (V, E)$ conexo e não direcionado, onde:
 - V é o conjunto de vértices (loais da vegetação).
 - E é o conjunto de arestas com pesos representando o custo de deslocamento.
- Localização inicial dos postos de brigadistas e pontos de coleta de água (considere que tais pontos possuem fontes infinitas de água).
- Parâmetros:
 - Capacidade de armazenamento dos caminhões de brigadistas.
 - Requisitos de água e equipes necessárias para extinguir um incêndio, podendo ser distintos dentre cada vértice.
- Ponto inicial do incêndio, diferente de um posto de de brigadistas.

3.2 Saídas

- Relatório contendo:
 - O número de vértices salvos em cada simulação.
 - Tempo necessário para conter o fogo (caso seja possível).
 - Caminhos percorridos pelas equipes de brigadistas.
 - Quantidade de água utilizada para apagar o fogo (caso seja possível).

4 Requisitos Técnicos

Os alunos deverão implementar a solução utilizando a linguagem de sua preferência e deverão aplicar algoritmos clássicos de grafos, tais como:

- **Busca em Largura (BFS)** ou **Dijkstra** para encontrar os caminhos mais curtos para os brigadistas.
- **Propagação baseada em BFS** para simular o avanço do fogo.
- **Estratégias para otimizar o deslocamento das equipes** considerando os custos das arestas, recursos necessários para apagar o fogo em cada vértice e capacidade dos caminhos.

5 Critérios de Avaliação

Os projetos serão avaliados com base nos seguintes critérios:

1. **Corretude da implementação (40%)**
 - O sistema simula corretamente a propagação do fogo e o deslocamento das equipes.
 - O relatório de saída reflete corretamente a eficácia do combate ao incêndio.
2. **Eficiência do Algoritmo (20%)**
 - Uso apropriado de estruturas de dados para otimizar tempo de execução.
 - Implementação de algoritmos adequados para resolver o problema.
3. **Qualidade do Código (20%)**
 - Código bem estruturado e documentado.
 - Uso adequado de modularização e boas práticas de programação.
4. **Análise e Discussão dos Resultados (20%)**
 - Relatório detalhado sobre as simulações realizadas.
 - Discussão sobre a eficiência do método proposto e possíveis melhorias.

6 Entrega

Os alunos deverão entregar:

- Código-fonte do sistema.
- Relatório explicativo com:
 - Descrição das decisões de implementação.
 - Análise dos resultados obtidos.
 - Possíveis melhorias e discussões sobre os desafios encontrados.