MC558 - Lista Avaliativa 2

Pedro Brasil Barroso - RA 260637

Universidade Estadual de Campinas
Instituto de Computação

Árvore Geradora Mínima

- **Problema escolhido:** *Acacias*, 8 pontos

- **Definição do problema:**

- **Entrada:** Um inteiro N, que representa o número de vértices, numerados de 1 a N, seguido de $N-1$ linhas. Cada linha i consiste em um inteiro não-negativo k, seguido de k pares de inteiros j e c_{ij}, tal que existe uma aresta não direcionada de i para j com peso c_{ij}.

- **Saída:** Uma linha com dois inteiros, tais que o primeiro representa o número de componentes conexas no grafo e o segundo, a soma dos custos das árvores mínimas de cada componente.

- **Solução:** O problema pode ser resolvido diretamente com pequenas alterações ao algoritmo de Kruskal visto em aula:

Algoritmo: AGM-KRUSKAL-ACACIAS(G, w)

```
A ← ∅
componentes ← |V[G]|
custo ← 0
para cada u ∈ V[G]
    MAKE-SET(u)
ordene as arestas em ordem não decrescente de peso
para cada (u, v) ∈ E[G] na ordem obtida
    se FIND-SET(u) ≠ FIND-SET(v)
        A ← A ∪ {(u, v)}
        UNION(u, v)
        custo ← custo + w(u, v)
        componentes ← componentes − 1
devolva custo, componentes, A
```

- **Complexidade:** Como visto em aula, o algoritmo de Kruskal para árvores geradoras mínimas apresenta complexidade $O(E \log E)$, onde E é o número de arestas do grafo.
  Como as operações adicionadas (instanciação e soma nas variáveis "componentes" e "custo") são realizadas em O(1), AGM-KRUSKAL-ACACIAS possui complexidade $O(E \log E)$.

- **Corretude:**
  Note que, como G pode ser desconexo, AGM-Kruskal-Acacias retorna uma floresta de árvores geradoras mínimas.
  Para provar a corretude, é necessário utilizar a seguinte invariante:

  > *No início de cada iteração, (1) pesos é a soma dos pesos das arestas em A e (2) componentes é o número de componentes conexas em* $G_A = (V, A)$.

  Base: No início do algoritmo, $A = \emptyset$ e $componentes = |V[G]|$; logo, a invariante é verdadeira.
  Hipótese: Suponha que a invariante é verdadeira no início da iteração i.
  Passo: Na iteração $i + 1$:
  Caso 1: A aresta (u, v) é adicionada a A.

  - (1) Então u e v são unidos em uma mesma componente conexa, reduzindo o número total de componentes em 1. Na linha 12, *componentes* é decrementada em 1.
  - (2) Além disso, o peso da aresta (u, v) é somado a *pesos* na linha 11.

  Caso 2: A aresta (u, v) não é adicionada a A.

  - (1) Neste caso, u e v já pertencem à mesma componente conexa, então *componentes* não é alterada.
  - (2) O peso da aresta (u, v) não é somado a *pesos*.

  Conclusão: A invariante é mantida em todas as iterações; logo, ao final do algoritmo, *componentes* é o número de componentes conexas em G_A e *custo* é a soma dos pesos das arestas em A.