

UNIVERSIDADE FEDERAL DA PARAÍBA  
CENTRO DE INFORMÁTICA

Disciplina: Análise e Projeto de Algoritmos (2019.2)

Professor: Bruno Petrato Bruck

Trabalho de implementação (Extra)

Seja  $G = (V, A)$  um grafo de entrada, onde  $V$  é o conjunto de vértices e  $A$  um conjunto de arestas. Seja ainda a função peso  $w : A \rightarrow \mathbb{R}$  associada a cada aresta  $(i, j)$ . O problema da Árvore Geradora Mínima consiste em encontrar um subconjunto  $T \subset A$ , tal que  $T$  forme uma árvore geradora e que a soma de suas arestas seja minimizada ( $\min \sum_{(i,j) \in T} w_{ij}$ ). Existem 2 algoritmos gulosos clássicos para a resolução desse problema, o algoritmo de Kruskal e o de Prim.

Neste trabalho você deve implementar **duas versões do algoritmo de Kruskal**. A primeira deve utilizar o algoritmo **HeapSort** (com a implementação das funções Max-Heapify e Build-Max-Heap como no livro texto da disciplina) para ordenar as arestas, enquanto a segunda deve utilizar o algoritmo **CountingSort**. Note que, tanto o HeapSort quanto o CountingSort devem ser implementados como parte do trabalho. Além da implementação, será necessário criar uma tabela comparando a média dos tempos de 10 execuções das duas versões do algoritmo para cada uma das instâncias de teste anexadas à descrição deste projeto.

**Observações:**

- Este trabalho **não é obrigatório**, mas vale 1 ponto extra na nota da prova P1.
- Os códigos devem ser enviados junto a um pdf com a tabela de resultados em arquivo compactado pelo SIGAA;
- O trabalho é **individual**;
- Será necessário **apresentação presencial** do trabalho ao professor.