Análisis de Algoritmos I

Profesora: Luz Gasca Soto Ayudantes: Bernal Cedillo Enrique Antonio y García Flores Jorge Luis Práctica 6: Prim y Kruskal Noviembre, 2017

1 Introducción

Esta práctica consiste en implementar los algoritmos que regresan un arbol generador de peso mínimo.

Específicamente, el algoritmo de Prim y el algoritmo de Kruskal.

2 Descripción

2.1 Entrada

El programa a implementar recibe como entrada en los argumentos de la linea de comandos:

- 1. Nombre del **archivo de texto** que contiene la información necesaria para construir la gráfica G. Esto es:
 - -En la primer linea, los **vértices** de la gráfica separados por ','.
 - -De la segunda linea en adelante, pares de vértices separados por ',' que indican las **aristas** de la gráfica y como ultimo dato su peso.
- 2. Nombre del algoritmo a utilizar.

Por ejemplo, para obtener el arbol generador de peso minimo usando Kruskal:

java SpanningTree Grafica.graph kruskal

2.2 Salida

El programa debe generar un nuevo archivo con la información del arbol generador de peso mínimo. (Con el mismo formato de la gráfica de entrada)

3 Detalles adicionales

La práctica puede ser implementada en ${\bf Java}$ o ${\bf Python}.$

Sin utilizar bibliotecas adicionales para el manejo de la gráfica.

(Estructuras auxiliares como Stacks, Queues, Arrays, etc... pueden ser utilizadas)

El algoritmo de Kruskal debe implementarse utilizando los conjuntos ajenos vistos en clase. Y el algoritmo de Prim con una cola de prioridades (Heap binario).

4 Fecha de Entrega

La fecha de entrega de la práctica es el día Viernes 1ro de Diciembre de 2017. Al correo enrique_bernal@ciencias.unam.mx Con asunto '[Practica06]ApellidopaternoPrimernombre'

4.1 Adjuntos

- -) Readme.txt, con nombre completo e instrucciones para compilar y ejecutar el programa.
- -) Archivos necesarios para la ejecución de su programa. (No se calificarán proyectos completos de IDEs)

Todo lo anterior en un archivo .zip/.rar/.tar.gz con el mismo nombre del asunto del correo.