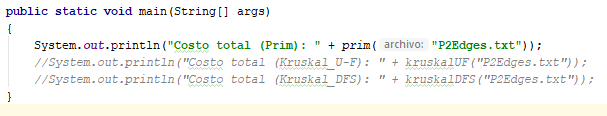
Rodrigo Garcia Mayo

A01024595

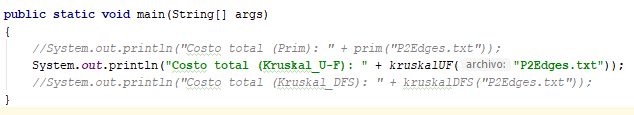
**Proyecto 2 – Árbol de expansión mínimo**

***Manual de usuario***

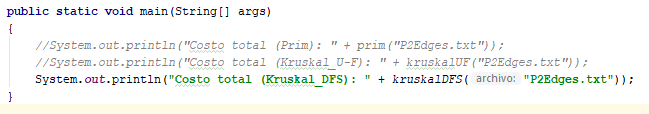
1. Descargar los archivos “Proyecto2\_MST” y “P2Edges.txt” y ponerlos en la misma carpeta
2. Abrir “Proyecto2\_MST” con un editor de texto, buscar la función “main” y seleccionar de la siguiente manera el método que quieras utilizar para obtener el MST (Prim, Kruskal utilizando DFS, Kruskal utilizando Union-Find):
   1. Para utilizar Prim: Modifique el main para que quede de esta manera



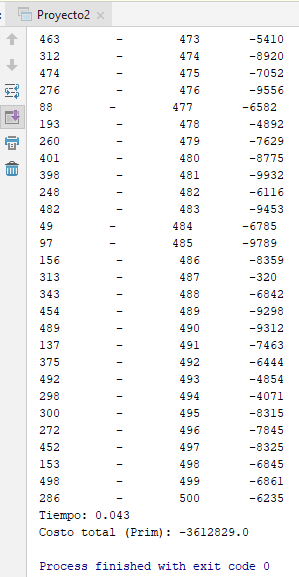
* 1. Para utilizar Kruskal con DFS: Modifique el main para que quede de esta manera



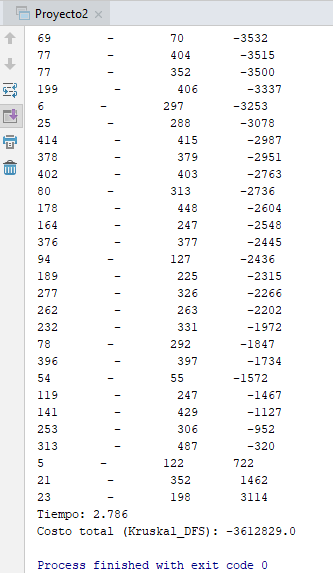
* 1. Para utilizar Kruskal con Union-Find: Modifique el main para que quede de esta manera



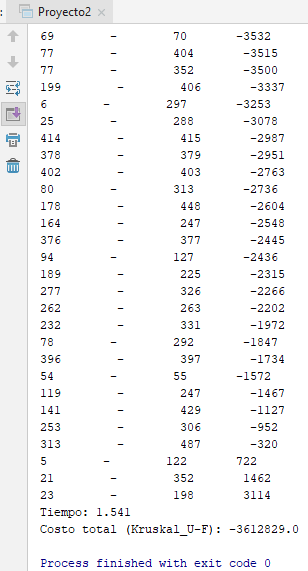
1. Una vez seleccionado el método que desea utilizar compilar y correr el archivo.
2. El programa ejecutara los siguientes resultados:
   1. Para Prim:



* 1. Para Kruskal con DFS:



* 1. Para Kruskal con Union-Find:



***Análisis de resultados:***

El mejor método que se encontró fue utilizando Prim con listas de adjacencia ya que el tiempo de ejecución requerido para encontrar el MST fue el menor. Esto se puede explicar con el orden que tiene Prim utilizando listas de adjacencia, siendo O(alog(m)).

El tiempo de ejecución de Kruskal\_DFS es el mayor ya que la estructura que el DFS tiene es de un orden mayor al de Union-Find. DFS tiene un orden de O(v+a) mientras que el Union-Find tiene un orden de O(alog(m) + nlog->(ordenamiento de aristas)).

Ahora, Prim y Kruskal con Union Find tienen el mismo orden. Pero para utilizar Kruskal con Union Find reorganice de forma ascendente el arreglo de aristas con el sort que utiliza java. Es la razón que encuentro por la que en este caso Prim sea más eficiente que Kruskal.

***Referencias:***

Khaled, O. (sf). Minimum Spanning Tree. Recuperado de <https://www.hackerearth.com/practice/algorithms/graphs/minimum-spanning-tree/tutorial/>

Geeks for Geeks. (2012). Prim’s Minimum Spanning Tree (MST). Recuperado de <https://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/>