

Ejercicios Lab08: Árbol de expansión mínimo

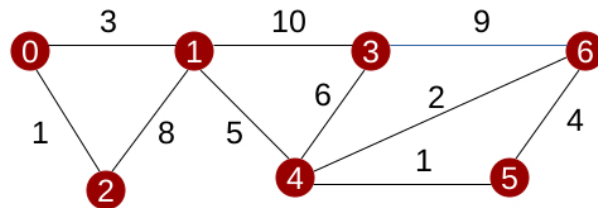
1. Encontrar un árbol de expansión mínimo

El algoritmo de Prim permite hallar el árbol minimal de cualquier grafo con pesos, no dirigido y conexo. Hay que seguir los siguientes pasos:

- Marcamos un nodo cualquiera, será el nodo de partida.
- Seleccionamos la arista de menor peso incidente en el nodo marcado anteriormente, y marcamos el otro nodo en el que incide.
- Repetimos el paso anterior siempre que la arista elegida enlace un nodo marcado y otro que no lo esté.
- El proceso termina cuando tenemos todos los nodos del grafo marcados.

Ejemplo

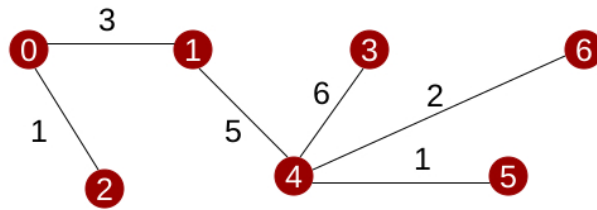
Vamos a construir el árbol de expansión mínimo para el siguiente grafo:



Siguiendo el algoritmo de Prim:

- Elegimos, por ejemplo, el nodo 0 y lo marcamos.
- Elegimos la arista con menor peso incidente en 0, la $(0, 2)$. Marcamos el otro nodo en el que incide, el 2.
- Elegimos la arista con menor peso incidente en un nodo marcado y otro que no lo esté, la $(0, 1)$. Marcamos el nodo no marcado, el 1.
- Elegimos la arista con menor peso incidente en un nodo marcado y otro que no lo esté, la $(1, 4)$. Marcamos el nodo no marcado, el 4.
- Elegimos la arista con menor peso incidente en un nodo marcado y otro que no lo esté, la $(4, 5)$. Marcamos el nodo no marcado, el 5.
- Elegimos la arista con menor peso incidente en un nodo marcado y otro que no lo esté, la $(4, 6)$. Marcamos el nodo no marcado, el 6.
- Elegimos la arista con menor peso incidente en un nodo marcado y otro que no lo esté, la $(4, 3)$. Marcamos el nodo no marcado, el 3.
- Finalizamos dado que tenemos marcados los 7 nodos del grafo.

Por tanto el árbol de expansión mínimo resultante sería:



2. Para realizar el ejercicio

Dispones del fichero `lab08_arbol.py`, donde tienes que implementar la función `prim_algorithm`. Dicha función recibe un grafo no dirigido conexo y con pesos, que son números reales, y devuelve un árbol de expansión mínimo del grafo.