

“Algoritmos”

***Algoritmo de Johnson***

Maestra:

Claudia Nayelli Sánchez

Alumno:

Emiliano Santos Galindo Janet 0188914

1 de diciembre del 2016

“Índice”

1.- Explicación del algoritmo 3

2.- Conclusión 5

“Explicación del algoritmo”

¿Qué es el algoritmo de Johnson?

El algoritmo de Johnson es un algoritmo de búsqueda en una gráfica.

¿Para qué sirve?

Sirve para encontrar el camino más corto entre todos los pares de nodos ordenados.

Se hace valer tomando otros dos famosos algoritmos, el algoritmo BellmanFord y el Dijkstra.

Su método de trabajo es que primero elimina los caminos negativos con BellmanFord, creando un nodo auxiliar, calcula los caminos más cortos a todos los nodos desde el nodo auxiliar, y después con la información obtenida cambia los pesos de todas las aristas para convertirlos en NO negativos.

Una vez hecho esto ejecuta Dijkstra para cada nodo.

Esto nos devolverá arreglos con los pesos de las rutas más cortas.

De cierta manera es como si usáramos Dijkstra pero sin la restricción de los pesos negativos en las aristas porque nos encargamos de eliminarlos.

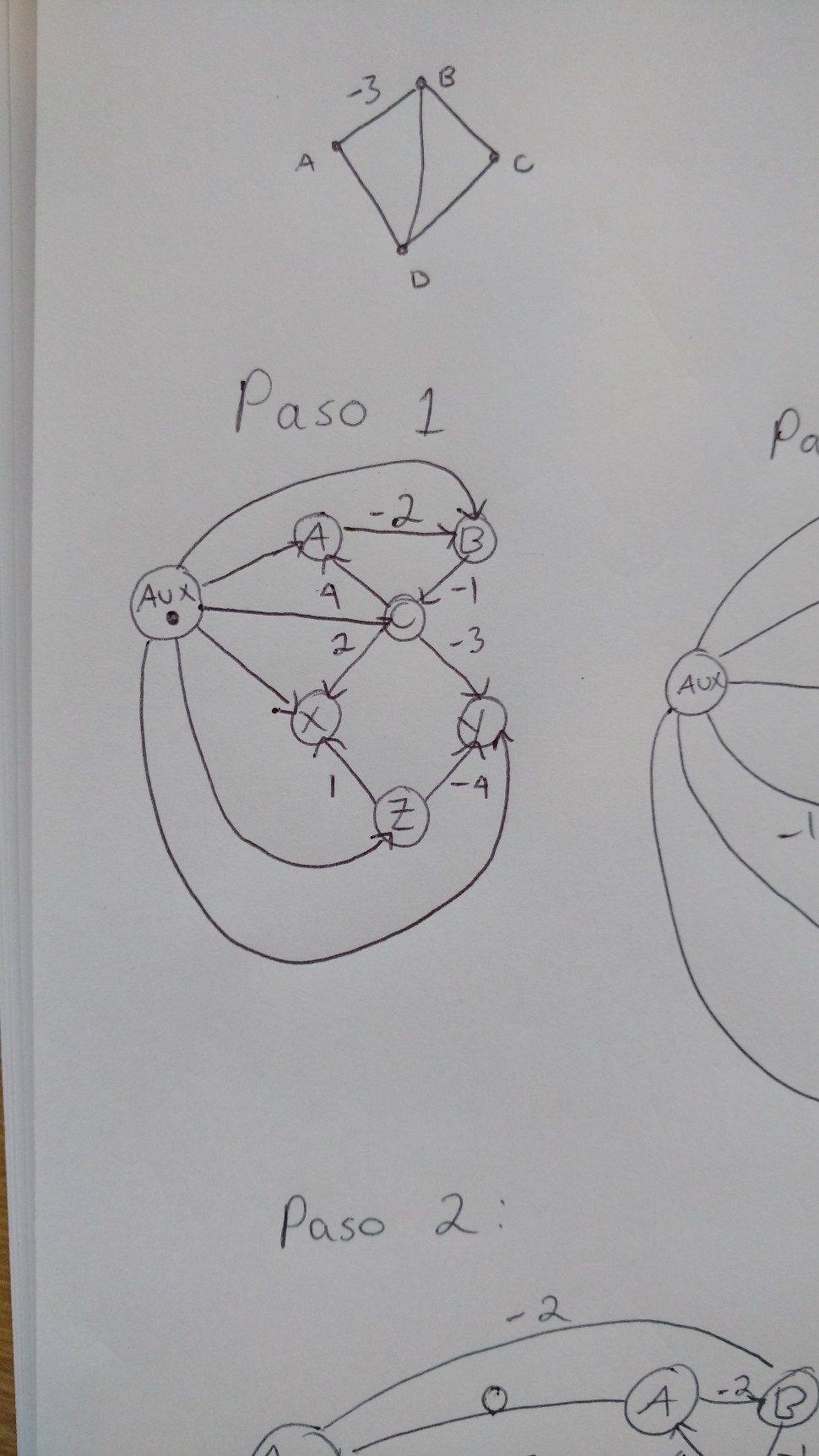
Algo muy importante es que esto se debe realizar con un Fibonacci Heap min-PriorityQueue.

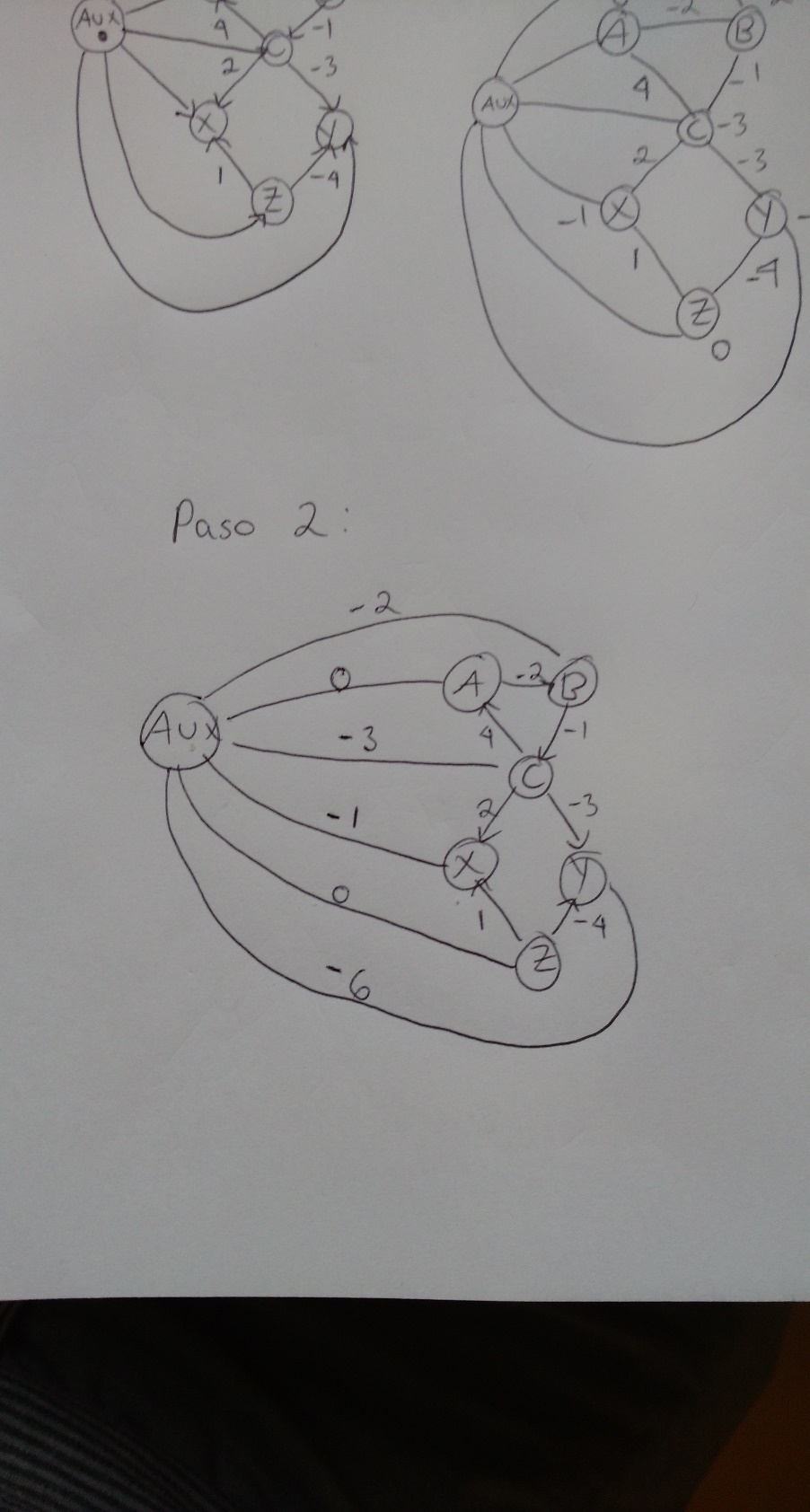
Su complejidad es de

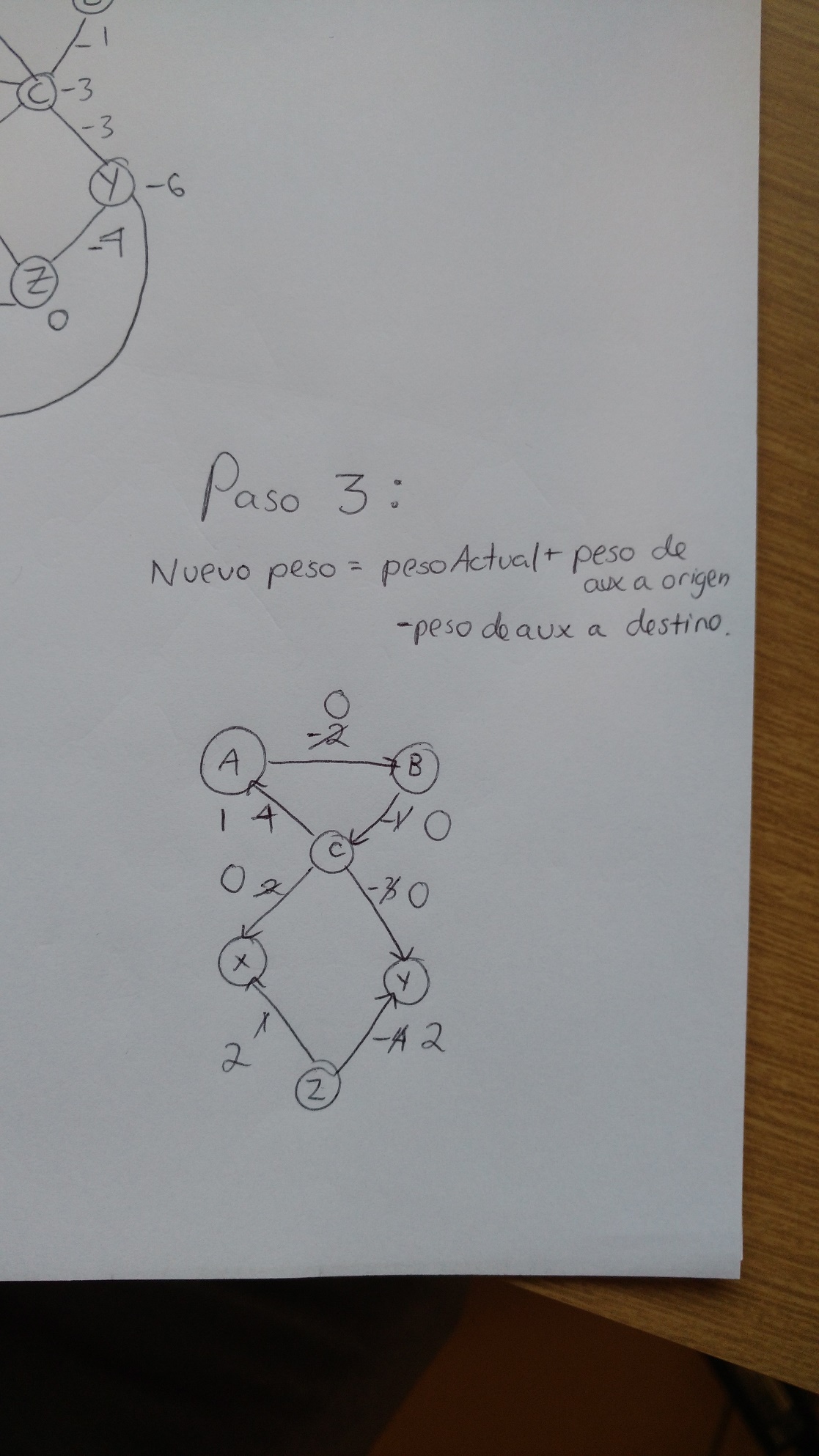
Su complejidad se volvería muy fea si el número de aristas es muy grande, es por eso que este algoritmo está pensado para gráficas con pocas aristas.

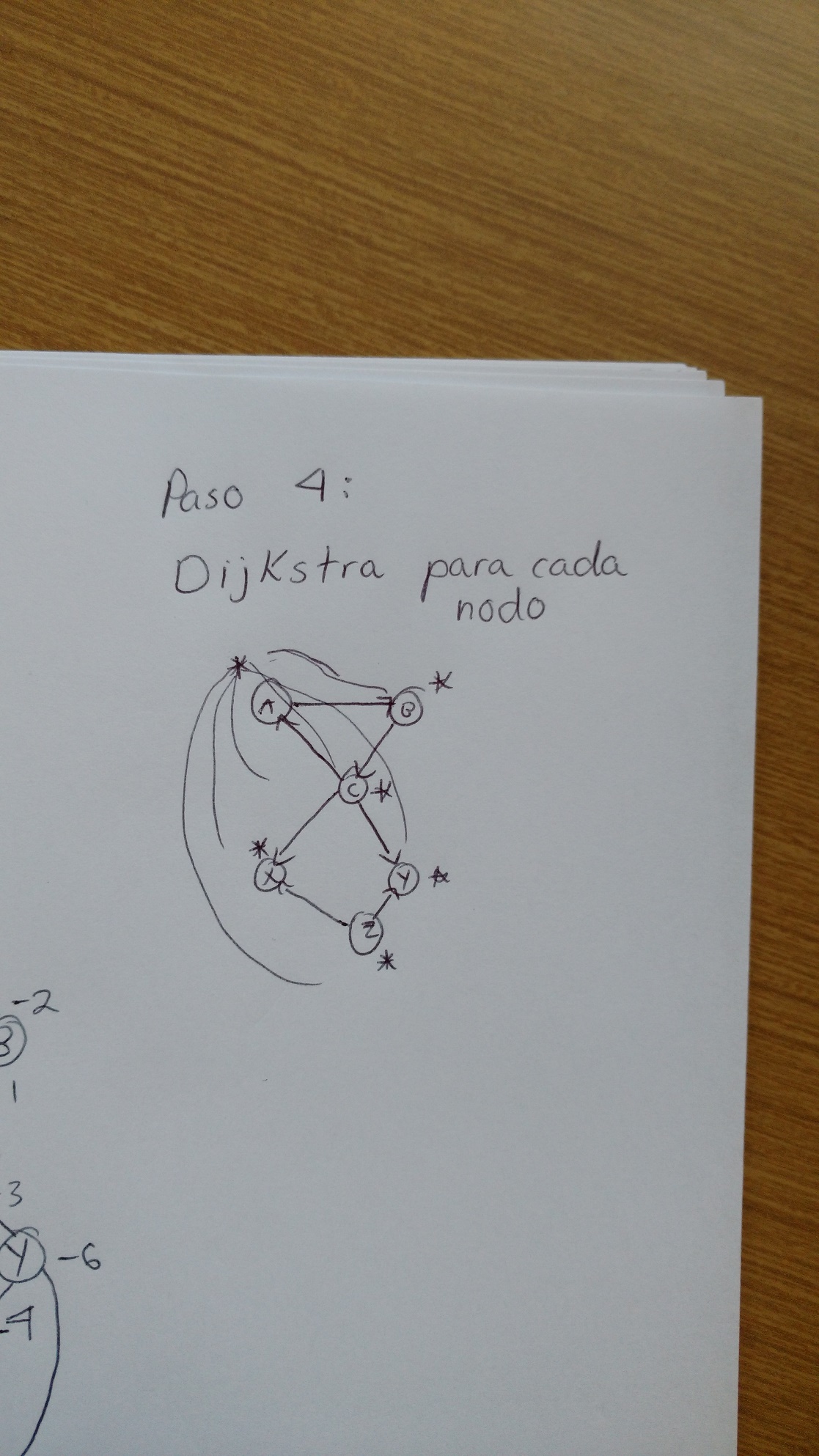
Sus pasos son los siguientes:

* 1.- Crear un nodo auxiliar y hacerlo adyacente a todos los demás
* 2.- Ejecutar BellmanFord del nodo auxiliar a todos los demás
* 3.- Actualizar pesos de la gráfica
* 4.- Correr Dijkstra V veces para cada nodo de la gráfica.









Comparando:

* Su complejidad es que es mejor que Floyd Warshall, que es
* Necesitas una estructura de datos especial (Fibonacci-heap min priority queue), en FloydWarshall solo necesitas un arreglo 2D.
* Floyd Warshall es un algoritmo por sí sólo, Jhonson’s no lo es, este necesita de dos algoritmos más

Conclusión

Este algoritmo es bueno, sin embargo es muy específico, ya que sólo es para graficas de pocas aristas.

Además sólo viene a mejorar la solución para soluciones ya existentes, lo que le resta importancia.

Fue intermedia su dificultad de implementación ya que son dos algoritmos en uno.