



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

M.I. Tista Garcia Edgar

Profesor:

Estructura de datos y Algoritmos II

Asignatura:

5

Grupo:

7

No de Práctica(s):

Calzada Martinez Jonathan Omar

Integrante(s):

*No. de Equipo de
cómputo empleado:*

35

05

No. de Lista o Brigada:

2019-2

Semestre:

19/03/2019

Fecha de entrega:

Observaciones:

CALIFICACIÓN: _____

Objetivo:

El estudiante conocerá e identificará las características necesarias para entender el algoritmo de búsqueda por profundidad de un grafo

Introduccion:

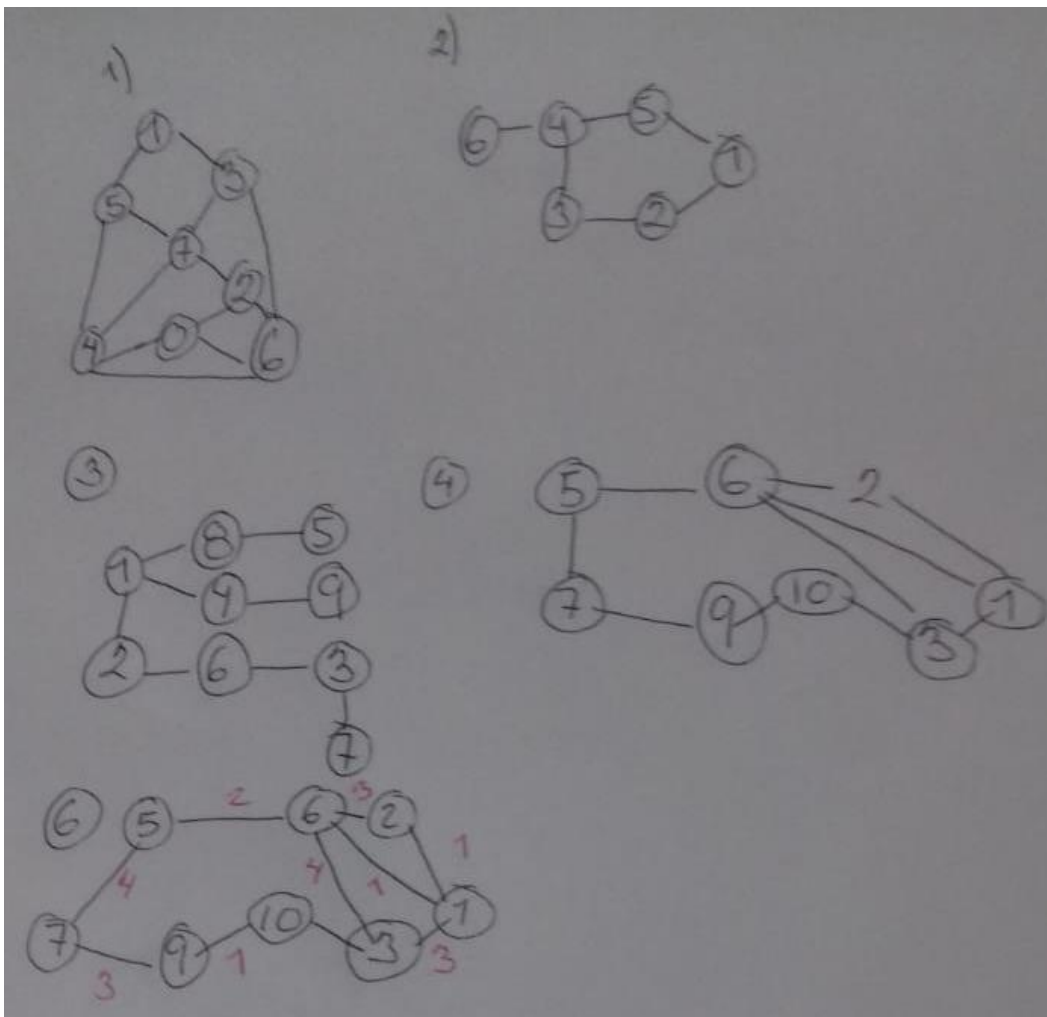
Una Búsqueda en profundidad (en inglés DFS o Depth First Search) es un algoritmo de búsqueda no informada utilizado para recorrer todos los nodos de un grafo o árbol (teoría de grafos) de manera ordenada, pero no uniforme. Su funcionamiento consiste en ir expandiendo todos y cada uno de los nodos que va localizando, de forma recurrente, en un camino concreto. Cuando ya no quedan más nodos que visitar en dicho camino, regresa (Backtracking), de modo que repite el mismo proceso con cada uno de los hermanos del nodo ya procesado.

Actividades:

Desarrollo:

- 1) Se elaboro un proyecto llamado practica 7, agregando la clase Graph realizada en la practica 6. y se agrego los siguientes metodos proporcionados en la clase graph.
- 2) Se implemento el grafo dado en el codigo dando el siguiente resultado

Se realizo el grafo que se pide que se realice y otros 3 grafos que se muestran en la siguiente imagen, junto con el que viene.



Se realizo 4 grafos el que se pidió que se realizara y los otros 3 que se especifican que se hagan y los resultados son los siguientes. El algoritmo hecho en este código funciona bien ya que haciendo el recorrido de manera Manuel como se explico en la clase de teoría .

```
DFS 1 desde vertice 1
1 3 7 5 4 0 2 6
```

```
DFS 2 desde vertice 1
1 5 4 6 3 2
```

```
DFS 3 desde vertice 1
1 8 5 4 9 2 6 3 7
```

```
DFS 4 desde vertice 1
1 3 10 9 7 5 6 2
```

```
DFS 5 desde vertice 1
Lista de Adyacencia del vertice 0
```

EJERCICIO 2

En la practica se pide usar el algoritmo ponderado que se pidió en la practica pasada pero no lo logre terminar entonces me entretuvo esa parte y solo logre terminar el algoritmo de grafos ponderados con BFS y se muestra en la parte de abajo, solo se logro eso por lo que no se pudo implementar el algoritmo de Kruskal.

```
DFS 5 desde vertice 1
Lista de Adyacencia del vertice 0
```

```
Lista de Adyacencia del vertice 1
1--- (1)--->2
1--- (2)--->3
1--- (1)--->6
```

```
Lista de Adyacencia del vertice 2
2--- (1)--->1
2--- (3)--->6
```

```
Lista de Adyacencia del vertice 3
3--- (2)--->1
3--- (4)--->6
3--- (3)--->10
```

Conclusiones:

En clase no había entendido bien el sentido de recorrer un grafo pero ya me di cuenta que es muy útil por ejemplo en un mapa para recorrer caminos por ejemplo del metro de la Ciudad o algo por el estilo.

Considero que el objetivo de la practica se obtuvo a medias ya que los conceptos se entendieron más la practica me costo un poco de trabajo y no pude realizarla de manera óptima.

En lo personal a la hora de realizar Kruskal fue en donde me atore y me llevo a tardarme demasiado ya que intentaba realizarlo desde cero el algoritmo pero al parecer mi lógica no era la correcta.

```
Grafica vista en clase
Lista de adyacencia del vertice 0
0
--> 1
--> 6
--> 4
```

```
Lista de adyacencia del vertice 1
1
--> 3
--> 0
--> 5
```

```
Lista de adyacencia del vertice 2
2
--> 7
--> 6
```

```
Lista de adyacencia del vertice 3
3
--> 1
--> 7
--> 6
```

```
Lista de adyacencia del vertice 4
4
--> 6
--> 0
--> 5
--> 7
```

```
Lista de adyacencia del vertice 5
5
--> 7
--> 1
--> 4
```

```
Lista de adyacencia del vertice 6
6
```

Conclusiones:

El algoritmo de DFS se esta de la misma manera que lo visto en clase, utulizando la recursividad.