

Facultad de Informática y Electrónica  
Escuela de Ingeniería en Sistemas



Integrantes: Códigos:

Juana Delgado 6078

Aldas Wilmer 6035

Cando Lizbeth 5951

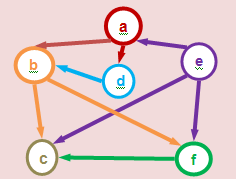
Pilataxi Claudio 6042

Romero Marcelo 6030

Semestre: Año Lectivo:

Tercero “A” Abril-Agosto 2016

Riobamba-Ecuador



Estructura De Datos

Trabajo Grupal

Tema**:** Exposición de grafos.

# INTRODUCCION

Típicamente, un grafo se representa gráficamente como un conjunto de puntos (vértices o nodos) unidos por líneas (aristas).

Desde un punto de vista práctico, los grafos permiten estudiar las interrelaciones entre unidades que interactúan unas con otras. Por ejemplo, una red de computadoras puede representarse y estudiarse mediante un grafo, en el cual los vértices representan terminales y las aristas representan conexiones (las cuales, a su vez, pueden ser cables o conexiones inalámbricas).

Prácticamente cualquier problema puede representarse mediante un grafo, y su estudio trasciende a las diversas áreas de las ciencias exactas y las ciencias sociales.

# OBJETIVOS

Objetivo General

Desarrollar e implementar el conocimiento abstracto para la resolución de problemas que simulan la vida real, mediante grafos.

Objetivos Específicos

* Desarrollar un esquema grafico para cada ejercicio planteado
* Especificar el desarrollo de la resolución de los ejercicios
* Implementar material de apoyo a la representación de soluciones
* Generar un ambiente de conocimiento concreto relativo al tema de grafos

# MARCO TEORICO

El primer artículo científico relativo a grafos fue escrito por el matemático suizo Leonhard Euler en 1736. Euler se basó en su artículo en el problema de los puentes de Königsberg. La ciudad de Kaliningrado, originalmente Königsberg, es famosa por sus siete puentes que unen ambas márgenes del río Pregel con dos de sus islas. Dos de los puentes unen la isla mayor con la margen oriental y otros dos con la margen occidental. La isla menor está conectada a cada margen por un puente y el séptimo puente une ambas islas. El problema planteaba lo siguiente: ¿es posible dar un paseo comenzando desde cualquiera de estas regiones, pasando por todos los puentes, recorriendo solo una vez cada uno y regresando al mismo punto de partida?

Abstrayendo este problema y planteándolo con la (entonces aún básica) teoría de grafos, Euler consigue demostrar que el grafo asociado al esquema de puentes de Königsberg no tiene solución, es decir, no es posible regresar al vértice de partida sin pasar por alguna arista dos veces.

De hecho, Euler resuelve el problema más general: ¿qué condiciones debe satisfacer un grafo para garantizar que se puede regresar al vértice de partida sin pasar por la misma arista más de una vez?

**Problema del camino más corto**

En la teoría de grafos, el problema del camino más corto es el problema que consiste en encontrar un camino entre dos vértices (o nodos) de tal manera que la suma de los pesos de las aristas que lo constituyen es mínima. Un ejemplo de esto es encontrar el camino más rápido para ir de una ciudad a otra en un mapa. En este caso, los vértices representarían las ciudades y las aristas las carreteras que las unen, cuya ponderación viene dada por el tiempo que se emplea en atravesarlas.

El problema del camino más corto puede ser definido para [grafos](https://es.wikipedia.org/wiki/Grafo) [no dirigidos](https://es.wikipedia.org/wiki/Grafo#Grafo_no_dirigido), [dirigidos](https://es.wikipedia.org/wiki/Grafo#Grafo_dirigido) o mixtos.

**Recorrido por profundidad**

Una Búsqueda en profundidad (en inglés DFS o Depth First Search) es un algoritmo que permite 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.

Análogamente existe el algoritmo de búsqueda en anchura (BFS o Breadth First Search).

**Recorrido por amplitud**

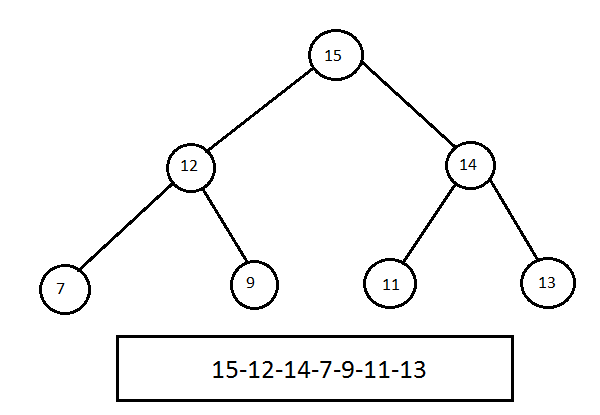
En Ciencias de la Computación, Búsqueda en anchura (en inglés BFS - Breadth First Search) es un algoritmo para recorrer o buscar elementos en un grafo (usado frecuentemente sobre árboles). Intuitivamente, se comienza en la raíz (eligiendo algún nodo como elemento raíz en el caso de un grafo) y se exploran todos los vecinos de este nodo. A continuación para cada uno de los vecinos se exploran sus respectivos vecinos adyacentes, y así hasta que se recorra todo el árbol.

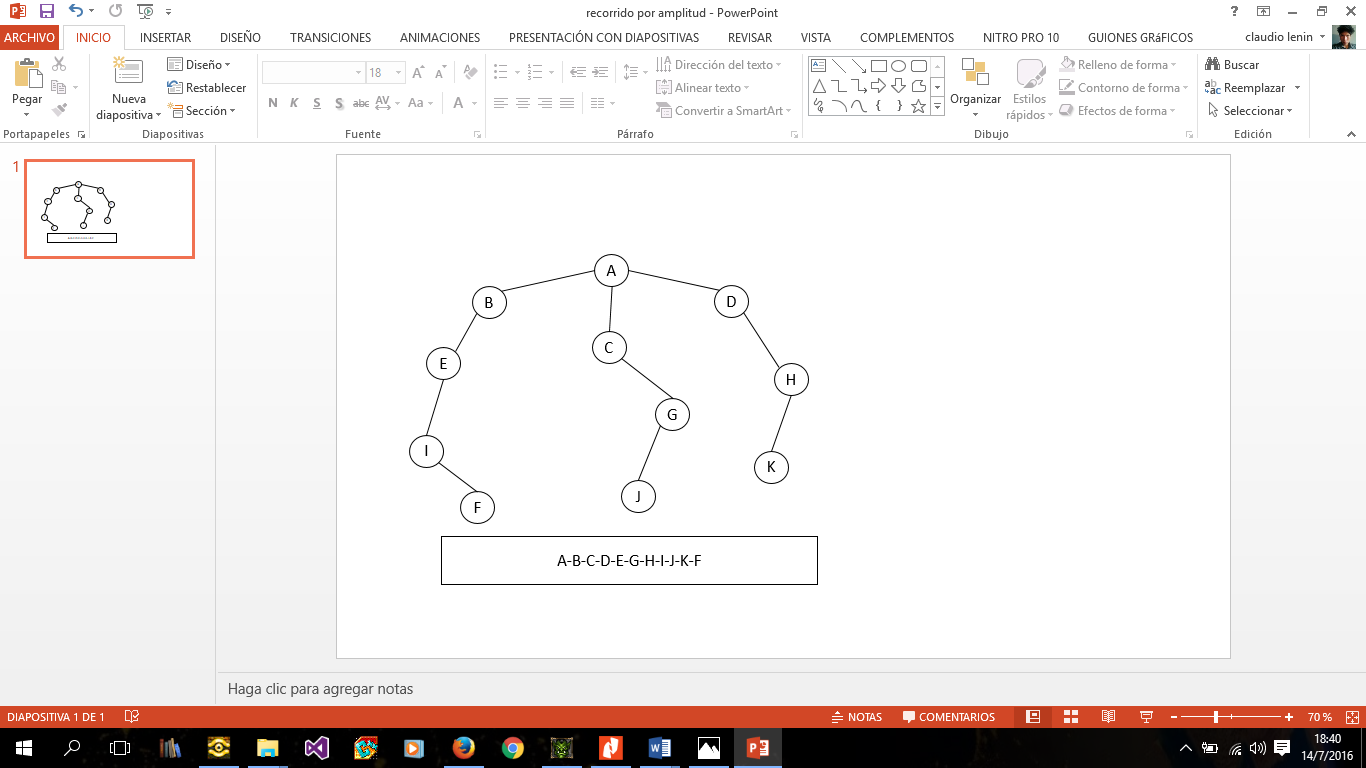
Formalmente, BFS es un algoritmo de búsqueda sin información, que expande y examina todos los nodos de un árbol sistemáticamente para buscar una solución. El algoritmo no usa ninguna estrategia heurística.

Si las aristas tienen pesos negativos aplicaremos el algoritmo de Bellman-Ford en alguna de sus dos versiones.

EJERCICIOS Y RESOLUCIONES

RECORRIDO POR AMPLITUD





DESDE J

J H D B F N K I C E M G A

Recorrido en Profundidad.

Adyacentes a 1: 2, 4, 8. El Recorrido seria: 1, 2

Adyacentes a 2: ~~1~~, 3, 4. El Recorrido seria: 1, 2, 3

Adyacentes a 3: ~~2,~~ 4, 5 El Recorrido seria 1, 2, 3, 4.

Adyacentes a 4: ~~1, 2, 3,~~ 7. El Recorrido seria: 1, 2, 3, 4, 7.

Recorrido en Profundidad.

Adyacentes a 5: ~~3, 6.~~ El Recorrido seria: 1, 2, 3, 4, 7, 6, 5

Adyacentes a 6: ~~5, 7~~. El Recorrido seria: 1, 2, 3, 4, 7, 6, 5.

Adyacentes a 7: ~~4, 6,~~ 9. El Recorrido seria: 1, 2, 3, 4, 7, 6, 5, 9.

Adyacentes a 9: ~~7,~~ 8. El Recorrido seria: 1, 2, 3, 4, 7, 6, 5, 9, 8.

**CAMINO MAS CORTO**

15

25

20

31

2

10

76

56

18

46

7

28

1

60

28

23

50

40

14

30

9

39

5

4

22

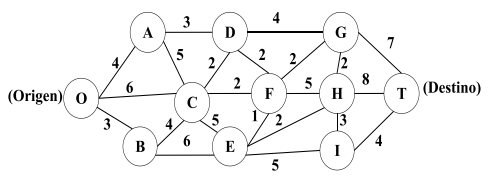
70

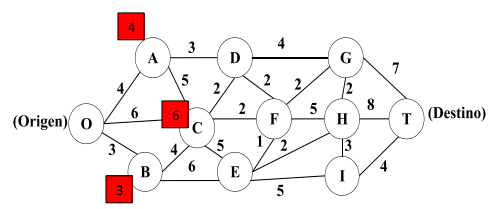
Desde I hasta L

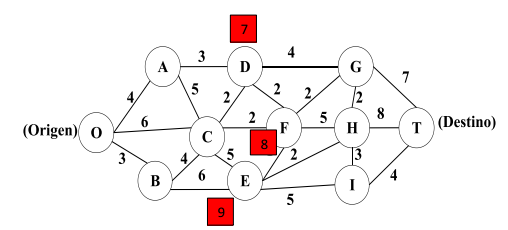
I + R + K + Q + B + O + L

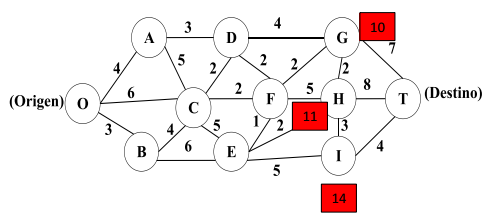
10 + 1 + 9 + 22 + 4 + 40 =86

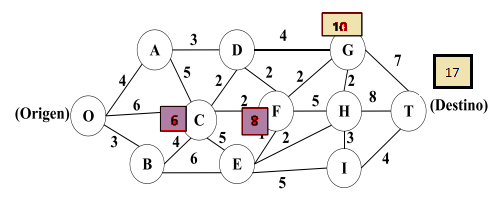
**Ejercicio del camino más cortó**











* **Camino más corto =** O + A + D + G + T
* **Resultado** = 4 + 3 + 4 + 7 = 17

**CONCLUSIONES**

* Se ha logrado desarrollar gráficamente los grafos planteados.
* Para cada ejercicio se ha encontrado una solución diferente
* Se ha representado las soluciones graficas con distintos programas
* Se ha logrado desarrollar todos los ejercicios de grafos sin dificultad

**RECOMENDACIONES**

* Usar el método más adecuado de ordenamiento para los distintos problemas
* Usar método gráficos para mejor compresión de los ejercicios