



Examen Sustitutorio  
Segundo Semestre Académico 2006

Curso : Estructura de Datos y Algoritmos II  
Grupo :  
Profesor(es) : Javier Cabrera Díaz y Sofía Álvarez Cárdenas  
Fecha : 15 de Diciembre  
Hora : 6:30pm a 8:30pm  
Duración de la prueba : 2 horas

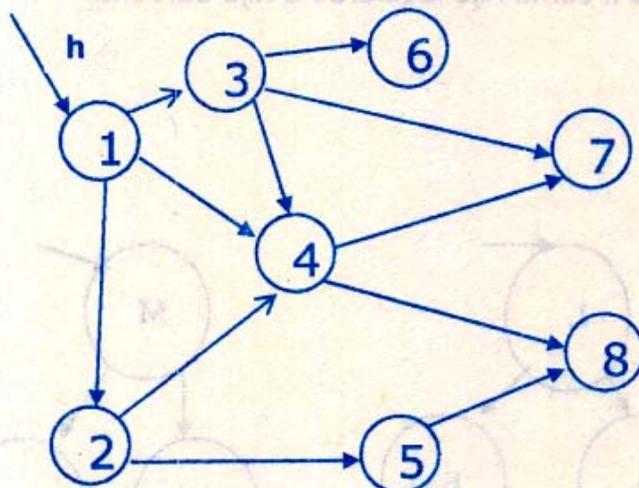
Nota: El examen es sin copias ni apuntes.

Está prohibido el préstamo de calculadoras, correctores, uso de celulares, consumo de bebidas, comidas y cigarrillos.

Se recomienda contestar las preguntas en el mismo orden en que aparecen. Utilice las clases cuya definición se anexa. No tiene que completar sus métodos, solo realice los métodos para responder las preguntas que se realizan.

I. Parte Práctica

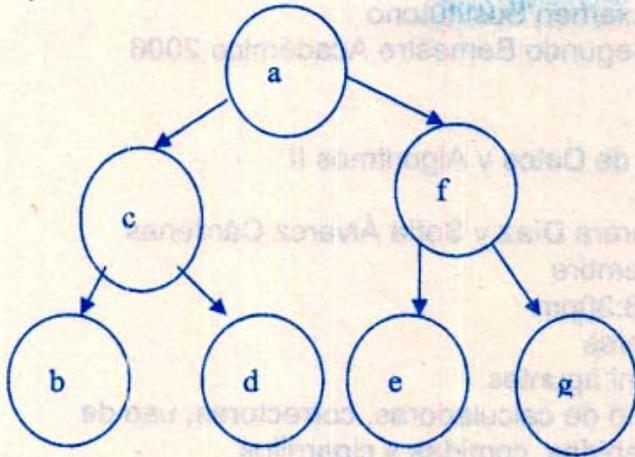
1. [3 PUNTOS]: Dado el siguiente grafo



- Construir la matriz de adyacencia
- Mostrar el recorrido de los nodos en profundidad. Partiendo de un nodo referenciado por h

c. Mostrar el recorrido en anchura. Partiendo de un nodo referenciado por h

2. [2 PUNTOS] Se tiene un árbol binario de búsqueda y se desea representarlo gráficamente en forma hilvanada



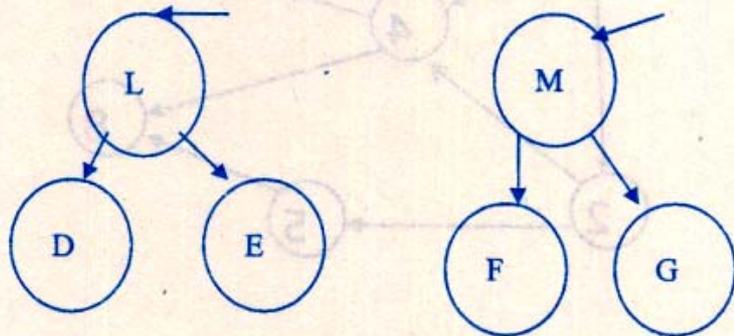
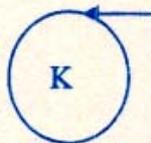
## II Parte Práctica

Realice los métodos en Java. Puede haber errores sintácticos o algunos elementos en pseudo código.

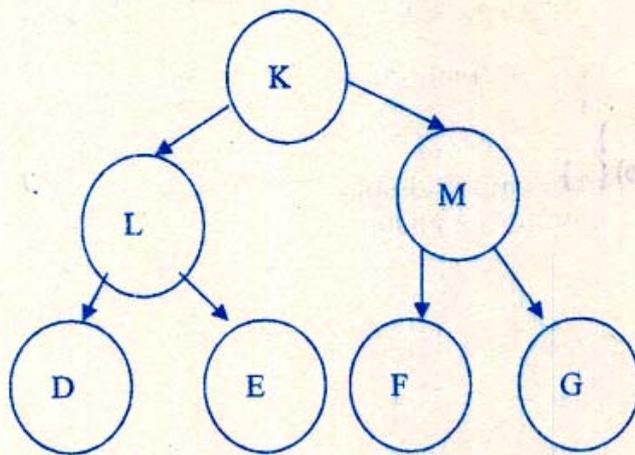
1. [4 puntos] Construir un árbol binario a partir de un nodo (k) cuya referencia es dato y de las raíces de dos subárboles (L y M). Las raíces L y M se colocaran al nodo k como hijo izquierdo e hijo derecho respectivamente.

Gráficamente se tiene:

Antes



Después



2. [5 puntos] Sea G un grafo dirigido y no ponderado con  $k$  vértices numerados desde 0 hasta  $k - 1$ , implementado mediante una matriz de adyacencia. Se desea saber:
  - a. Si existe un único vértice con grado de entrada 0, que se llamará **raíz**
  - b. Si todos los vértices del grafo, distintos de la **raíz**, tienen grado de entrada 1.
  
3. [6 puntos] Se tiene un fichero de **cadenas de caracteres**, cada cadena contiene la dirección de una vivienda y el distrito, separados por comas (**cada cadena ocupa 30 caracteres**) y se desea obtener una copia de este fichero donde aparezcan solo las cadenas de viviendas del distrito **La Molina**.

#### Anexo

##### Para árbol binario

```

public class Nodo {
    private int llave; //la llave del nodo
    private String inf; //la informacion del nodo
    private Nodo hijoIzq; //el hijo izquierdo del nodo
    private Nodo hijoDer; //el hijo derecho del nodo
    public Nodo(int id, String cc)//Constructor
    {}
    public void mostrar()
    {
        System.out.print ("{}");
        System.out.print (llave);
        System.out.print (",");
        System.out.print (inf);
        System.out.println ("");
    }
}
  
```

```

public Nodo dameHijoIzq () { }
public Nodo dameHijoDer () { }
    public int dameLLave () { }
    public String dameInf () { }
public void ponerHijoIzq (Nodo hijo) { }
    public void ponerHijoDer (Nodo hijo) { }
}
public class Arbol {
    public Nodo raiz;
    public Arbol() {
        raiz = null;
    }
}

```

### Sobre grafo

```

public class Vertice
{ private String llave;
  private boolean fueVisitado;
  public Vertice(String llav)
  {
  }
  public String dameLlave()
  {
  }

  public boolean yaFueVisitado()
  {
  }
} //fin de clase vértice

public class AdyMatriz {

    private int Max_Verts =20;
    private Vertice verticeLista[];//arreglo de vertices
    private int adyMat[][];//matriz de adyacencia
    private int nVertices;//numero actual de vertices

    public AdyMatriz()
    {
    }

    public void insertarVertice(String llav)
    {
    }

    public void insertarArco(int inicio, int fin)
    {
    }

    public void muestraVertice(int v)

```

```
{  
}  
public void muestraMatriz()  
{  
}
```

### **Sobre ficheros**

File(String S)

File f= new File("MiArchivo");

```
File f1,f2,f3;  
f1 = new File("Pedro");  
f2 = new File("Juan");  
f3 = new File("Diego");  
f1.delete();  
f2.renameTo(f3);
```

FileOutputStream fs = null;

```
fs = new FileOutputStream("texto.txt");  
fs.write(buffer,0,nbytes);
```

FileInputStream fe = null;

```
fe = new FileInputStream("texto.txt");  
nbytes = fe.read(buffer,0,81
```

---

**Nota:** El lunes 18 a las 2pm será la devolución del examen en sala de profesores B203 o en la Escuela de Ingeniería Informática B206