

Ingeniería Civil Informática *

Estructuras de Datos

Pauta Certamen #1

Otoño 2017

Problema 1

```
import java.util.*;
class Serie {
//nodo de la lista
private class Nodo {
    int val;
    Nodo elSiguiente ;
    Nodo(int val, Nodo els) {
        this.val = val;
        elSiguiente=els;
    }
}
//-----
private Nodo laCabeza;
int size; // Elementos almacenados en la lista
Serie(){
    laCabeza=null;
    size=0;
}
//-----
public void Insertar(int x) {
    size++;
    if (laCabeza==null) laCabeza=new Nodo(x,null);
    else if (x <= laCabeza.val) laCabeza= new Nodo(x,laCabeza);
    else {
        Nodo la= laCabeza;
        Nodo lt = la;
        for(;la!=null && la.val < x; la = la.elSiguiente)lt = la;
```

*Universidad del Bío-Bío / Departamento de Ciencias de la Computación y Tecnologías de la Información

```

        lt.elSiguiente=new Nodo(x,la);
    }
}
//-----
int Minimo() throws NoSuchElementException {
    if (laCabeza == null) throw new NoSuchElementException("No existe minimo");
    Nodo t = laCabeza;
    laCabeza= laCabeza.elSiguiente;
    size--;
    return t.val;
}
//-----
int Size() {
    return size;
}
//-----
void Imprimir() {
    Nodo t = laCabeza;
    for (; t!=null; t = t.elSiguiente) System.out.print(t.val + " - ");
}
} // Fin clase Serie

```

Problema 2

```

class tinyInt{
    byte n;
    tinyInt() {
        n =0 ;
    }
    //-----
    public void Set(byte i, byte b) {
        if(i==1) {
            n = (byte) (n | 0xf0);
            b = (byte) (b << 4);
            b = (byte) (b | 0x0f);
            n = (byte) (n & b);
        }
        else {
            n = (byte) (n | 0x0f);
            b = (byte) (b | 0xf0);
            n = (byte) (n & b);
        }
    }
    //-----
    public byte Get(byte i) {
        if(i==1) {
            byte t = (byte) n;

```

```

        return (byte) (t>>>4);
    }
    else {
        byte t = (byte) (n & 0x0f);
        return t;
    }
}
//-----
public void Add(byte i, byte v) {
    byte t = (byte) ((Get(i)) + v);
    Set(i, t);
}
}

```

Problema 3 Estos métodos van dentro de la clase ArbolNario

```

public int nodosxNivel(int i){
    return nodosxnivel(root, 0, i);
}
//-----
private int nodosxnivel(NodoArbol n, int cnt, int i)
{if (n ==null || cnt> i) return 0;
    else if(i==cnt) return 1+ nodosxnivel(n.hermano, cnt,i);
    else return (nodosxnivel(n.hijo,cnt+1,i)+ nodosxnivel(n.hermano, cnt,i));
}

```

Problema 4 Estos métodos van dentro de la clase Abb

```

//-----
public int Sucesor(int clave) {
    return sucesor(laRaiz, clave, -1);
}
//-----
private int sucesor(NodoAbb n, int clave, int suc) throws NoSuchElementException {
    if(n==null) throw new NoSuchElementException("No hay sucesor");
    else {
        if(n.elemento == clave) {
            if(n.rchild !=null) suc = Buscar_minimo(n.rchild);
            return suc;
        }
        else if(clave > n.elemento) return sucesor(n.rchild, clave,suc);
        else {
            suc = n.elemento;
            return sucesor(n.lchild,clave,suc);
        }
    }
}

```

```
}  
//-----  
private int Buscar_minimo(NodoAbb raiz) {  
    if(raiz.lchild==null) return raiz.elemento;  
    else return Buscar_minimo(raiz.lchild);  
}
```