UNIVERSIDAD ORT
Uruguay

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

EVALUACION	Solución de Examen	**GRUPO**		**FECHA**	19/05/2023
MATERIA	Algoritmos 1				
CARRERA	Analista Programador / Analista en Tecnologías de la Información				
CONDICIONES	**- Puntaje: 0/100** **- Duración: 2 horas** **- Sin material**				

Ejercicio 1 (20 ptos)

Se desea realizar la implementación de un TAD Lista doblemente encadenada

a) Defina la clase Nodo necesaria para que pueda contener datos de tipo entero.
b) Dentro de la lista Lista, defina sus atributos (se asumen disponibles los métodos gets y sets) e implemente la operación agregarInicio(int dato).

a)

```
public class NodoDoble {

    private int dato;
    private NodoDoble sig;
    private NodoDoble ant;

    //Constructor
    public NodoDoble(int n){
        this.dato=n;
        this.sig=null;
        this.ant=null;
    }
```

b)

```
public class Lista implements IListadoble{

    NodoDoble inicio;

    public Lista() {
            inicio = null
    }

    @Override
    public void agregarInicio(int dato) {
            NodoDoble nodo = new NodoDoble(dato);
            if(this.esVacia()){
                    this.setInicio(nodo);
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

```
                }
                else {
                        nodo.setSig(this.inicio);
                        this.inicio.setAnt(nodo);
                        this.setInicio(nodo);
                }
        }
}
```

Ejercicio 2 (20 ptos)

Solución:

```
public static void imprimirNroSocios(Lista l,String nom) {

      NodoSocio actual= l.getInicio();
      int pos = 1;
      while (actual != null) {
        if (actual.getDato().getNombre().equals(nom)) {
           System.out.println(pos);
        }
        actual=actual.getSiguiente();
pos++;
      }
   }
```

Ejercicio 3 (20 puntos)

```
public int valoresPares( (Pila p) {

       int valor = 0;

       if (!p.estaVacia()) {

          if ((int) (p.obtenerTope()) % 2 == 0) {
             valor = (int) p.obtenerTope();
          }

          p.desapilar();
          return valor + sumarPilaRec(p);
       }
       else{
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

```
            return 0;
        }

    }
```

Ejercicio 4 (20 ptos)

```
public void eliminarDuplicados(Lista l) {

    Nodo actual = l.getInicio();
    if (actual != null |&& actual.getSiguiente() && null) {

        // Recorremos la lista
        while (actual.getSiguiente() != null) {
            // Si el siguiente nodo tiene el mismo valor, lo eliminamos
            if (actual.getDato() == actual.getSiguiente().getDato()) {
                actual.setSiguiente(actual.getSiguiente().getSiguiente());
            } else {
                // Si no tiene el mismo valor, avanzamos al siguiente nodo
                actual = actual.getSiguiente();
            }

        }
    }
}
```

Ejercicio 5 (20 ptos)

c) Implementar un algoritmo recursivo que imprima los elementos de la diagonal opuesta de una matriz.

```
public void diagonalOpuesta (int largoFila, int largoCol, int[][] m) {

        System.out.print("Elementos de la diagonal opuesta: ");
        imprimirDiagonalOpuesta(m, 0, m[0].length - 1);
    }

    public  void imprimirDiagonalOpuesta(int[][] matriz, int i, int j) {
        // Caso base: llegamos al final de la diagonal
        if (i != matriz.length ) {

    System.out.print(matriz[i][j] + " ");
                imprimirDiagonalOpuesta(matriz, i + 1, j - 1);
        }
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

```
}
```

d) Realizar el diagrama de llamadas para el ejemplo dado

Diagrama de llamadas:

```
I=0, j=2
Se imprime 3
    ImprimirDiagonalOpuesta(m,1,1)
        I=1  , j=1
        Se imprime 3
        ImprimirDiagonalOpuesta(m,2,0)
            I=0 , j=0
            Se imprime 6
            ImprimiDiagonalOpuesta(m, 3, -1)
                Retorna.
```