

# TRABAJO PRÁCTICO – ESTRUCTURAS DINÁMICAS

## Ejercicio Nº 1: LISTAS DOBLEMENTE ENCADENADAS

Responder las siguientes preguntas:

- ¿Qué hace el siguiente trozo de programa?

```
nuevo (p);  
si (p = null) entonces escribir ("Error");  
sino  
    Escribir ("Ingrese dato");  
    Leer (dato);  
    *p.valor:=dato;  
    *p.proximo:=null;  
    *p.anterior:=null;  
    prim:=p;  
    ult:=p;
```

- ¿Qué relación existe entre este trozo de programa y el anterior?

```
nuevo (p);  
mientras (p<>Null) ^ (i<cant) hacer  
    Escribir ("Ingrese dato");  
    Leer (dato);  
    *p.valor:=dato;  
    *ult.proximo:=p;  
    *p.anterior:=ult;  
    *p.proximo:=null;  
    ult:=p;  
    i:=i+1;  
    Nuevo (p);  
fin mientras
```

- ¿Cuál de las siguientes opciones permite mostrar los elementos de una lista desde el final?

**a)** escribir ("Mostrar los elementos de la lista a partir del primer elemento de la lista");  
p:=prim;  
mientras (p<> Null) hacer  
    escribir (\*p.valor);  
    p:=\*p.proximo;  
fin mientras

**b)** escribir ("Mostrar los elementos de la lista a partir del último elemento de la lista");  
p:=ult;  
mientras (p<> Null) hacer  
    escribir (\*p.valor);  
    p:=\*p.anterior;  
fin mientras

- ¿Qué operación realiza el trozo de algoritmo en la siguiente imagen?

```
nuevo (p);  
si (p = null) entonces escribir ("Error");  
sino  
    t:=prim;  
    escribir ("Ingrese dato");  
    Leer (dato);  
    *p.valor:=dato;  
    *prim.anterior:=p;  
    *p.proximo:=prim;  
    *p.anterior:=null;  
    prim:= p;  
fin si
```

### Ejercicio Nº 2:

Indicar si el siguiente trozo de algoritmo corresponde a la operación de apilar o encolar

Algoritmo

i=0;

Escribir ("Ingresar cantidad de elementos de la lista");

Leer (cant);

nuevo (p);

si (p = null) entonces escribir ("Error");

sino

    Escribir ("Ingrese dato");

    Leer (dato);

    \*p.dato:=dato;

    \*p.proximo:=null;

    prim:=p;

```
i:=i+1;  
nuevo (p);  
Mientras (p<>Null)  $\wedge$  (i<cant) hacer  
    Escribir ("Ingrese dato");  
    Leer (dato);  
    *p.valor:=dato;  
    *p.proximo:=prim;  
    prim:=p;  
    i:=i+1;  
    Nuevo (p);  
fin mientras
```