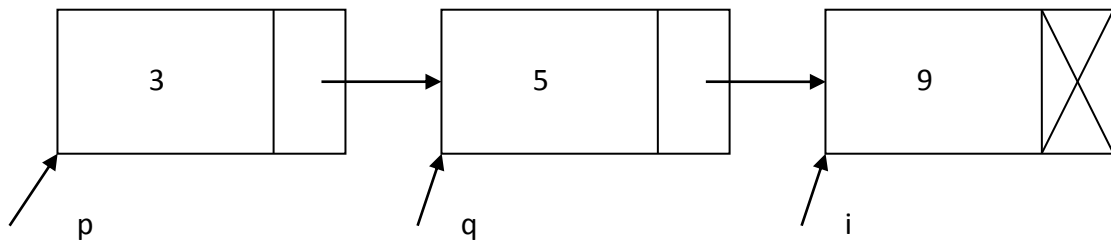


Guía de ejercicios

Utilizar la siguiente declaración en los ejercicios del 1 al 3.

```
typedef struct lista {  
    int info;  
    struct lista * sig;  
}nodo;
```

1. Mostrar lo que harían las siguientes instrucciones en la lista simplemente enlazada:



a) $p = p \rightarrow \text{sig}$

b) $q = p$

c) $i = p \rightarrow \text{sig}$

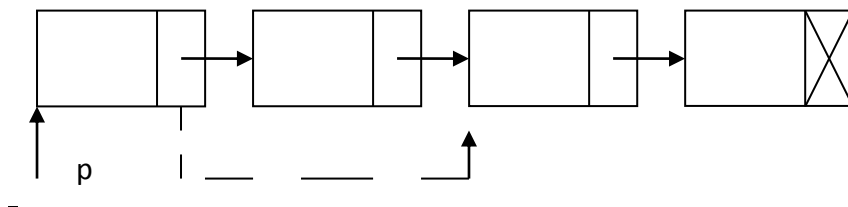
d) $p \rightarrow \text{info} = i \rightarrow \text{info}$

e) $p \rightarrow \text{info} = q \rightarrow \text{sig} \rightarrow \text{info}$

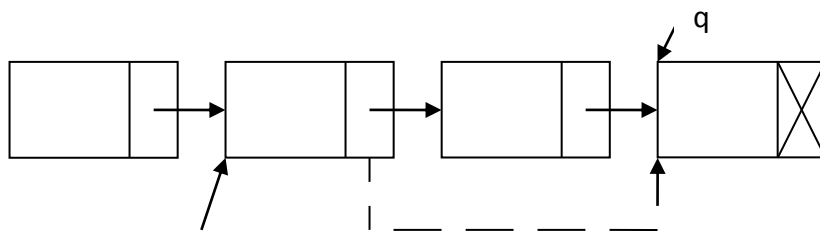
f) $i \rightarrow \text{sig} = p$

2. Escribir una sentencia para efectuar el cambio indicado en la línea de puntos.

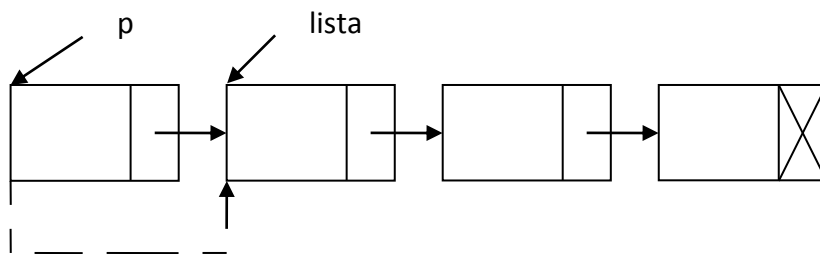
a)



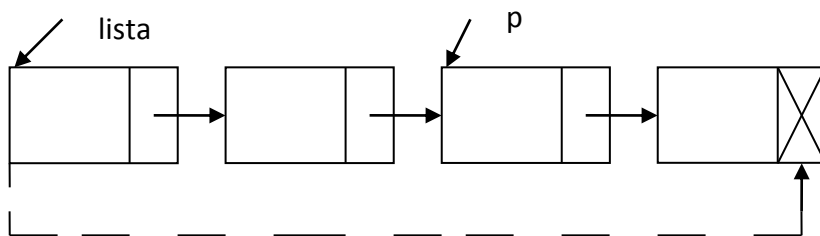
b)



c)



d)



3. Mostrar lo que escriben los siguientes segmentos:

a) crear (p);

crear (q);

p→info=5;

q→info=6;

p=q;

p→info=1;

emitir (p→info);

emitir (q→info);

b) crear(p);

p→info=3;

crear(q);

p→info= q→info;

q→info=0;

emitir(p→ info);

emitir (q→info);

4. Dada la siguiente lista simplemente enlazada que contiene la siguiente información:

DNI: entero positivo de hasta 8 dígitos.

Nombre: cadena de 15 caracteres

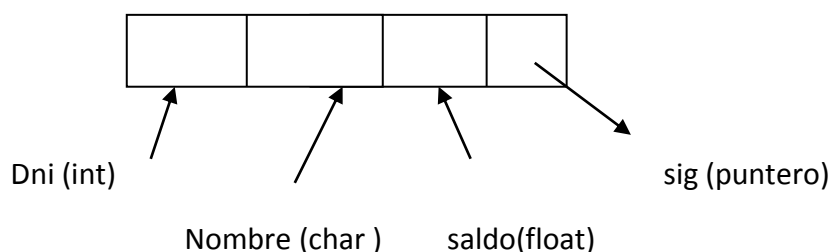
Tipo de cuenta: carácter (C,E,A)

Saldo: real

Siguiente: puntero al próximo elemento de la lista.

Desarrollar un procedimiento que busque en la lista un nodo cuyo DNI sea igual a uno dado. El mismo devolverá un puntero al nodo hallado o NULL, si no existiera tal nodo.

5. Dada la siguiente estructura:



a) Crear y mostrar la lista. La carga de datos termina con dni=0. La carga de datos se hace en forma ascendente por dni.

b) Generar otra lista con todos los saldos negativos.

6. Dada la siguiente estructura:

```
typedef struct lista {  
    int codmat;  
    char nombre [15];  
    struct lista * sig;  
} nodo;
```

- a)** Crear y mostrar la lista (la carga finaliza cuando `codmat=0`). Se cargan todos los códigos de materias iguales seguidos y ordenados.
- b)** Mostrar los alumnos que pertenecen a una materia.
- c)** Mostrar el código de materia que cuenta con más alumnos.