

El código fuente de la función InsertarCabezaLista:

```
void InsertarCabezaLista(Nodo** cabeza, Item entrada)
{
    Nodo *nuevo ;
    nuevo = (Nodo*)malloc(sizeof(Nodo)); /* asigna nuevo nodo */
    nuevo -> dato = entrada; /* pone elemento en nuevo */
    nuevo -> siguiente = *cabeza; /* enlaza nuevo nodo al frente de la lista */
    *cabeza = nuevo; /* mueve puntero cabeza y apunta al nuevo nodo */
}
```

### Caso particular

La función InsertarCabezaLista actúa también correctamente si se trata el caso de añadir un primer nodo o elemento a una lista vacía. En este caso, y como ya se ha comentado cabeza apunta a NULL y termina apuntando al nuevo nodo de la lista enlazada.

### Ejercicio 18.1.

Crear una lista de números aleatorios e insertar los nuevos nodos por la cabeza de la lista. Una vez creada la lista, se han de recorrer los nodos para mostrar los números pares.

La función InsertarCabezaLista() añade un nodo a la lista, siempre como nodo cabeza. El primer argumento es un puntero a puntero porque tiene que modificar la variable cabeza, que es a su vez un puntero a Nodo. La función NuevoNodo() reserva memoria para un nodo, asigna el campo dato y devuelve la dirección del nodo creado.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define MX 99
#define randomize (srand(time(NULL)))
#define random(num) (rand()%(num))
typedef int Item;
typedef struct Elemento
{
    Item dato;
    struct Elemento* siguiente;
}Nodo;

void InsertarCabezaLista(Nodo** cabeza, Item entrada);
Nodo* NuevoNodo(Item x);
void main()
{
    Item d;
    Nodo *cabeza,*ptr;
    int k;

    cabeza = NULL; /* Inicializa cabeza a lista vacía */
    randomize;
    /* El bucle termina cuando se genera el número aleatorio 0 */
    for (d = random(MX); d; )
    {
```

```
InsertarCabezaLista(&cabeza,d);
d = random(MX);
}
/* Ahora se recorre la lista para escribir los pares */
for (k = 0,ptr=cabeza; ptr; )
{
    if (ptr->dato%2 == 0)
    {
        printf("%d ",ptr->dato);
        k++;
        printf("%c", (k%12?' ':'\n')); /* cada 12 datos salta de línea */
    }
    ptr = ptr->siguiente;
}
printf("\n\n");
}
void InsertarCabezaLista(Nodo** cabeza, Item entrada)
{
    Nodo *nuevo ;
    nuevo = NuevoNodo(entrada);
    nuevo -> siguiente = *cabeza; /* enlaza nuevo nodo al frente de la lista */
    *cabeza = nuevo; /* mueve puntero cabeza y apunta al nuevo nodo */
}
Nodo* NuevoNodo(Item x)
{
    Nodo *a ;
    a = (Nodo*)malloc(sizeof(Nodo)); /* asigna nuevo nodo */
    a -> dato = x;
    a -> siguiente = NULL;
    return a;
}
```

### Inserción de un nuevo nodo que no está en la cabeza de lista

La inserción de un nuevo nodo no siempre se realiza al principio (en cabeza) de la lista. Se puede insertar en el centro o al final de la lista. El siguiente ejemplo muestra cómo insertar un nodo entre dos nodos de la lista, y se escriben, paso a paso, las sentencias.

### Ejemplo 18.4.

Se desea insertar un nuevo elemento 75 entre el elemento 25 y el elemento 40 en la lista enlazada 10, 25, 40.

