

Un nodo puede tener dos apuntadores que se declaran en la clase en la forma:

```
Public class Lista Doble
```

```
{// Datos privados
```

```
    ListaDoble uno; // apuntador uno
```

```
    ListaDoble dos; // apuntador dos
```

```
    // Constructor y todos los demás métodos
```

```
}
```

Esa clase a modo lógico en la literatura se presenta de dos formas:

Uno: nodo → datos uno dos →

Dos: nodo → uno datos dos

Usaremos la clase motocicleta declarada en Java, para generar una lista de liga doble de motos. El inicio de esa lista será el apuntador principio, que al inicio será null. Principio→null.

Generamos un nodo con una moto, en la forma:

Moto1=new Motocicleta ("BMW", 500,"TURISMO"); el nodo generado es:

Principio→null

Moto1 →uno, "BMW", 500,"TURISMO", dos →null
null ←

El problema es insertar ese nodo al inicio de la lista, hay dos casos, la lista es vacía o no está vacía.

Caso 1. La lista es vacía, se tiene:

_principio→null
Moto1 →uno, "BMW", 500,"TURISMO", dos →null
null ←

Caso 2. La lista no es vacía, se tiene entonces:

_principio→nodo1→nodo2→nodo3→null
null ←
Moto1 → uno, "BMW", 500,"TURISMO", dos → null
null ←

```
If (principio==null) principio=nodo;
```

```
else nodo.sig=principio;
```

```
principio.ante=nodo
```

```
principio=nodo;
```

Otro problema es insertar ese nodo al final de la lista, en éste caso hay que localizar primero cual es el último nodo de la lista.

Supongamos que la lista de liga doble es:

Principio → datos1 ↔ datos2 ↔ datos3 ↔ datos4 ↔ datos5 → null

Observemos que el apuntador siguiente, sig, de cada nodo no es null en un nodo intermedio, pero es null para el último nodo. Entonces se puede usar la condición `nodo.sig=null` para buscar el último nodo.

Luego una rutina para buscar el último nodo es:

```
Motocicleta aux=new Motocicleta()
```

```
Motocicleta ultimo=new Motocicleta()
```

```
Si(principio=null) escribe lista vacía, no se puede insertar nodo
```

```
else aux=principio
```

```
    while(aux!=null)
```

```
    { si(aux.sig==null) { ultimo=aux; aux=aux.sig; }
```

```
      else aux=aux.sig;
```

```
} Motocicleta nodo=new Motocicleta(marca, cil, tipo);
```

```
Ultimo.sig=nodo; nodo.ante=ultimo;
```