

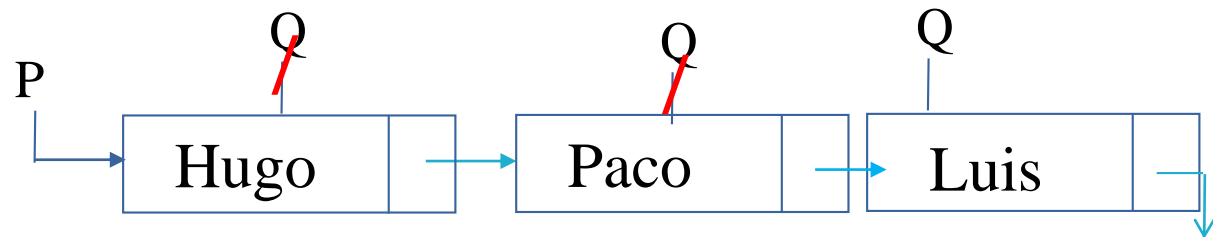
Unidad 3. Estructuras lineales estáticas y dinámicas

- ▶ **Listas enlazadas simples**

Recorridos en listas lineales de enlace simple:

- ▶ El recorrido en una lista enlazada es un proceso sencillo; se inicia por el primer nodo y avanza al siguiente hasta llegar final de la lista.

$Q=P$



- ▶ Se requiere un recorrido en una lista enlazada para imprimir su contenido, buscar un dato, contar los nodos de la lista, obtener una copia, etc. .

Recorridos en listas lineales de enlace simple:

Ejercicio en clase:

Diseña un algoritmo para imprimir el contenido de una lista



Algoritmo imprimelista()

- 1.Inicio
2. $aux = P$
- 3.Repetir mientras ($aux \neq \text{Nulo}$)
 Escribe $aux.info$
 $aux = aux.liga$
4. Fin_ciclo_paso_3
- 5.Fin

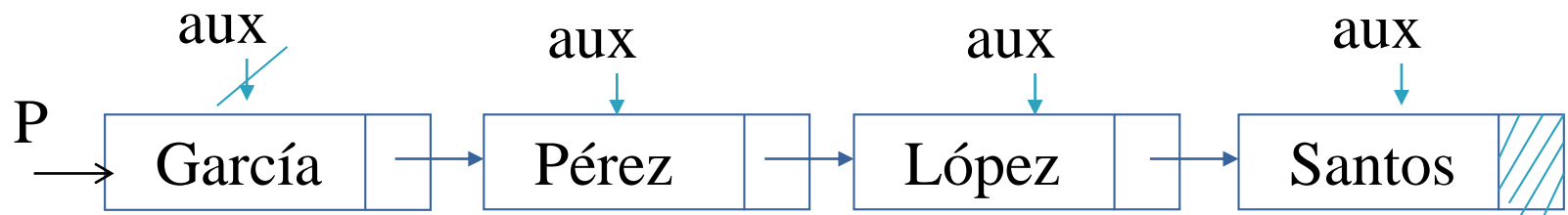
Salida:

García

Pérez

López

Santos



Práctica

Implementa el tda lista dinámica simple y sus operaciones básicas:

Inserta

- * Al inicio de la lista.
- * Al final de la lista.
- * Antes/después de un nodo dado como referencia.

Eliminar

- * Eliminar el primer nodo.
- * Eliminar el último nodo.
- * Eliminar un nodo con información X.

Imprime lista

- * inicio a fin

Buscar valor x en lista



Implementación de la clase Lista de enlace simple

- ▶ Especificación, UML, codificación
- ▶ Se requieren dos clases de objetos:
 - Clase Nodo
 - Clase ListaSimple

Implementación

clase Nodo

Especificación (representación lógica)

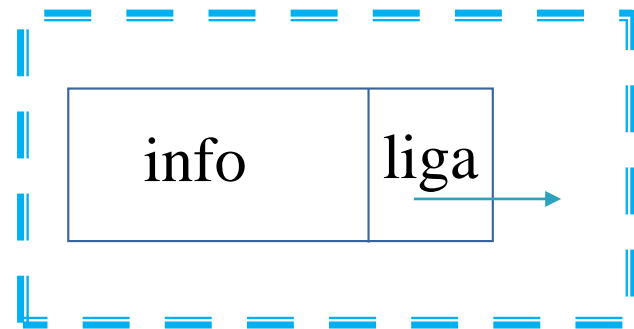
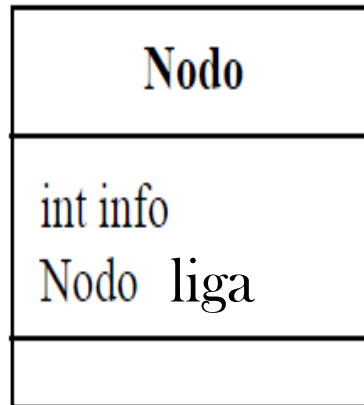


Diagrama UML



Codificación

JAVA

```
class Nodo {  
    int info;  
    Nodo liga;  
}
```

Implementación

clase ListaSimple

Especificación

(representación lógica)

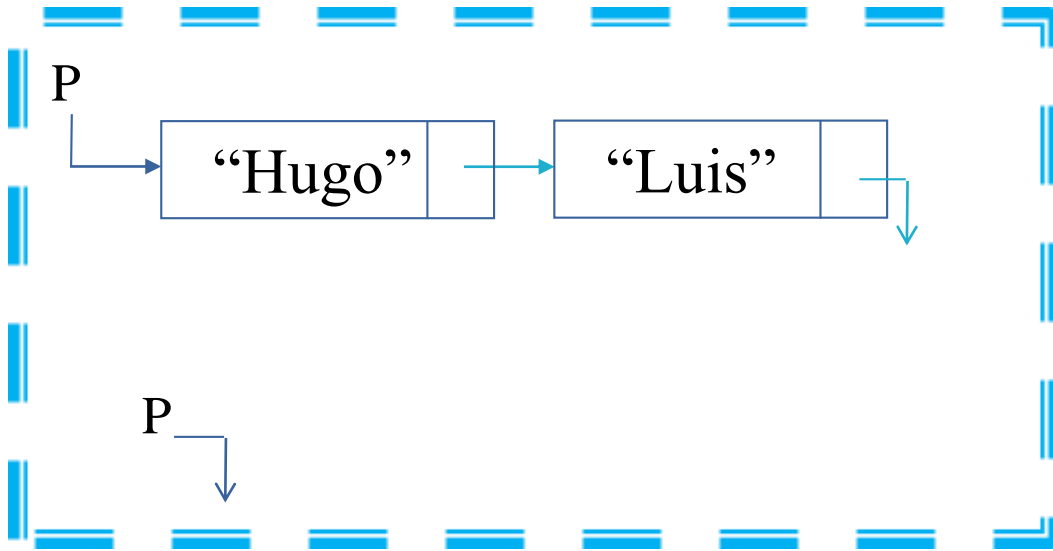


Diagrama UML

ListaSimple

Nodo p

Insertainicio()

Insertafinal()

InsertaAntes()

Eliminaprimer()

Eliminafinal()

EliminaNodoX()

Imprimelista()

BuscaNodo()

- ▶ Código en Java

Las siguientes diapositivas muestran una implementación sencilla del tda lista lineal de enlace simple.

```
1
2 package listas;
3 public class Nodo {
4     int info;
5     Nodo liga;
6 }
7
8
9
```

<default confi...     

ListaSimple.java x Listas.java x Nodo.java x

Source History        

```
2  package listas;
3
4  public class ListaSimple {
5      Nodo p;
6
7      public ListaSimple() { //crea una lista vacía
8          p=null;
9      }
10     public void insertaInicio(int n) {
11         Nodo q=new Nodo();
12         q.info=n;
13         q.liga=p;
14         p=q;
15     }
16     public void insertaFinal(int n) {
17         Nodo aux;
18         aux=p;
19         while(aux.liga!=null) {
20             aux=aux.liga;
21         }
22         Nodo q=new Nodo();
23         q.info=n;
24         q.liga=null;
25         aux.liga=q;
26     }
```

```
27
28 public void Eliminarprimero() {
29     Nodo q=p;
30     if (q.liga==null){
31         p=null;
32     }
33     else {
34         p=q.liga;
35     }
36     q=null;
37 }
38 public void Eliminaultimo() {
39     Nodo q;
40     Nodo t=null;
41     if (p.liga==null){
42         p=null;
43     }
44     else {
45         q=p;
46         while (q.liga!=null){
47             t=q;
48             q=q.liga;
49         }
50         t.liga=null;
51         q=null;
52     }
53 }
54 public void Imprimelista() {
55     Nodo q=p;
56     while (q!=null){
57         System.out.println(q.info);
58         q=q.liga;
59     }
60 }
```

}

ListaSimple.java x Listas.java x Nodo.java x

```
Source History
1 package listas;
2 import java.util.*;
3
4 public class Listas {
5
6     public static void main(String[] args) {
7         Scanner sc=new Scanner (System.in);
8         ListaSimple L=new ListaSimple();
9         System.out.println("Inserta el valor de inicio");
10        int n=sc.nextInt();
11        L.insertaInicio(n);
12        System.out.println("LISTA INICIAL");
13        L.Imprimelista();
14        L.insertaFinal(8);
15        L.insertaInicio(2);
16        System.out.println("LISTA: ");
17        L.Imprimelista();
18        System.out.println("SE HAN ELIMINADO ALGUNOS ELEMENTOS");
19        L.Eliminaprimer();
20        L.Eliminaultimo();
21        System.out.println("LISTA FINAL");
22        L.Imprimelista();
23    }
24
25 }
```

Fuentes de información

- ▶ Estructuras de Datos, Tercera Edición, Osvaldo Cairo y Silvia Guardati, Mc Graw Hill.
- ▶ <http://sabadosinformaticos.blogspot.mx/2013/08/estructura-de-datos-cairo-y-guardati.html>
- ▶ <https://kesquivel.files.wordpress.com/2010/08/listas-circularesv2010.pdf>