

Clase 22: Diccionarios con listas enlazadas

Uno de los grandes problemas de los arreglos es su tamaño fijo.

Implementaremos un Diccionario mediante el uso de listas enlazadas, una estructura de datos dinámica, en el sentido que puede cambiar su tamaño durante el transcurso del programa.

1 Nodo

Un nodo es un objeto donde se guarda la definición del diccionario:

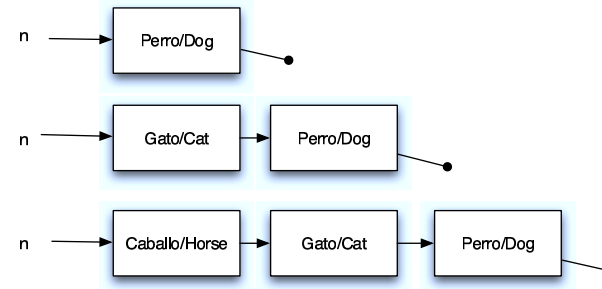
```
class Nodo {
    String llave, valor;
    Nodo(String l, String v) {
        llave=l;
        valor=v;
    }
}
```

2 Diccionario

La siguiente clase Diccionario permite guardar 0 o 1 definición:

```
class Diccionario {
    Nodo definicion;
    Diccionario() {
        definicion=null;
    }
    void put(String l, String v) {
        Nodo n=new Nodo(l,v);
        definicion=n;
    }
    Nodo get(String l) {
        if (definicion==null)
            return null;
        if (definicion.llave.equals(l))
            return definicion.valor;
        return null;
    }
    void remove(String l) {
        if (definicion==null)
            return;
    }
}
```

Figure 1: Lista de nodos



```
if (definicion.llave.equals(l))
    definicion=null;
}
```

Este diccionario tiene una ventaja: su tamaño es variable y una desventaja, a lo más puede almacenar una definición.

3 Un nodo enlazado

Haremos una nueva versión de Nodo, donde además del par valor/llave, guardaremos en cada Nodo una referencia a un siguiente Nodo. Esto permite crear listas de Nodos, como se ve en la figura.

```
class Nodo {
    String llave, valor;
    Nodo siguiente;
    Nodo(String l, String v, Nodo s) {
        llave=l;
        valor=v;
        siguiente=s;
    }
}
```

En el caso que un nodo sea el último, el valor de su campo siguiente es null.

Para crear la lista de la figura, el código es

```
Nodo n=new Nodo("Perro","Dog",null);
Nodo p=new Nodo("Gato","Cat",n);
Nodo q=new Nodo("Caballo","Horse",p);
```

4 Un diccionario con listas enlazadas

La siguiente clase Diccionario usa una lista enlazada para permitir mantener un diccionario que crezca:

```
class Diccionario {
    Nodo definicion;
    Diccionario() {
        definicion=null;
    }
    void put(String l, String v) {
        Nodo n=new Nodo(l,v,definicion);
        definicion=n;
    }
    Nodo get(String l) {
        Nodo n=definicion;
        while (n!=null) {
            if (n.llave.equals(l))
                return n.valor;
            n=n.siguiente;
        }
        return null;
    }
}
```

5 Problema propuesto

Implementa el método remove con una lista enlazada