

Ejercicio 5. Dado un fichero de palabras no significativas, y un fichero de títulos, se pretende crear otro fichero de índices de palabras en el contexto (Kwic).

Por ejemplo, sea el fichero de palabras no significativas:

el la los las un una
de entre para por
y en desde hasta
sobre tras del
cómo al ir

Y el fichero de títulos (con un título por línea)

El libro de la selva
Un león extraño
Cómo ir al cine en verano
Un amigo del león en la selva
El verano en un libro extraño
El extraño amigo del león

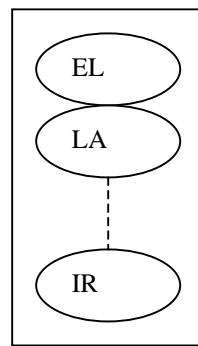
Produce el siguiente resultado:

amigo del león | El extraño
amigo del león en la selva | Un
cine en verano | Cómo ir al
extraño amigo del león | El
extraño | El verano en un libro
extraño | Un león
león | El extraño amigo del
león en la selva | Un amigo del
león extraño | Un
libro de la selva | El
libro extraño | El verano en un
selva | El libro de la
selva | Un amigo del león en la
verano | Cómo ir al cine en
verano en un libro extraño | El

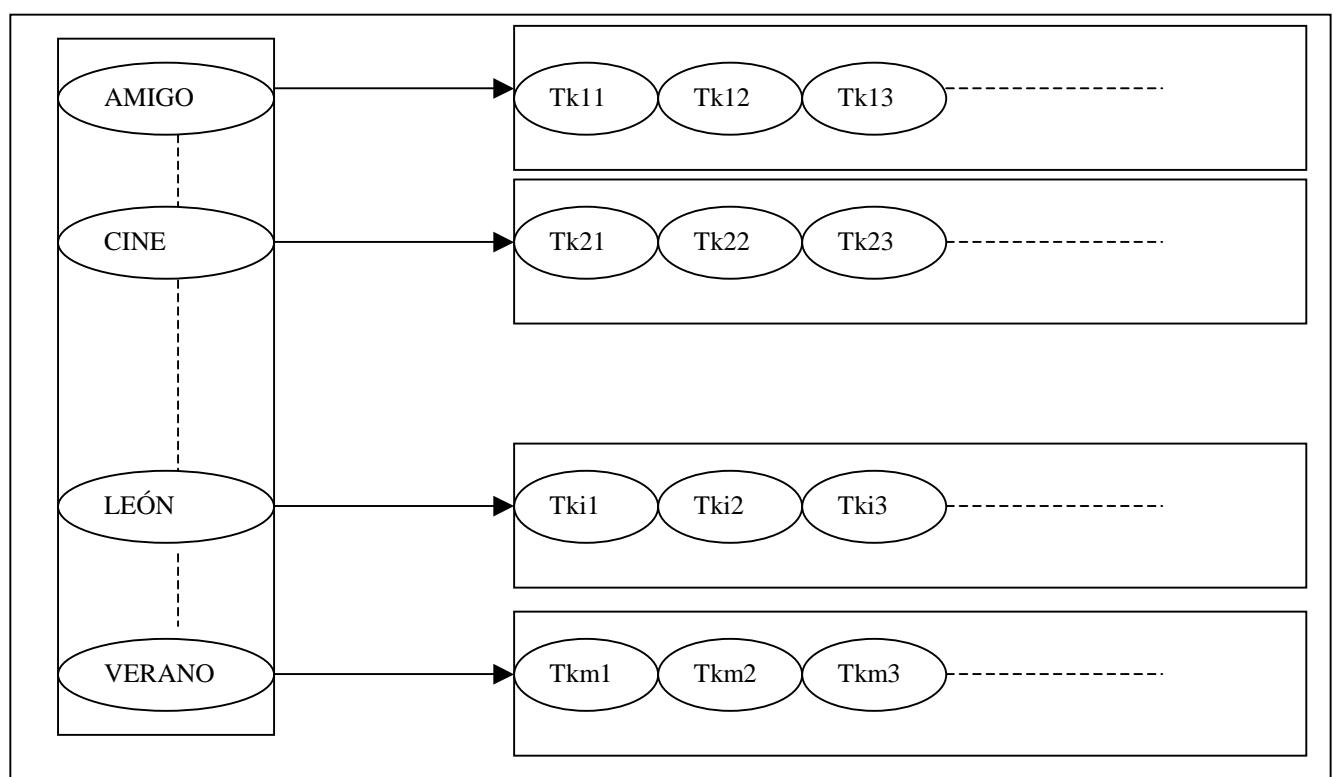
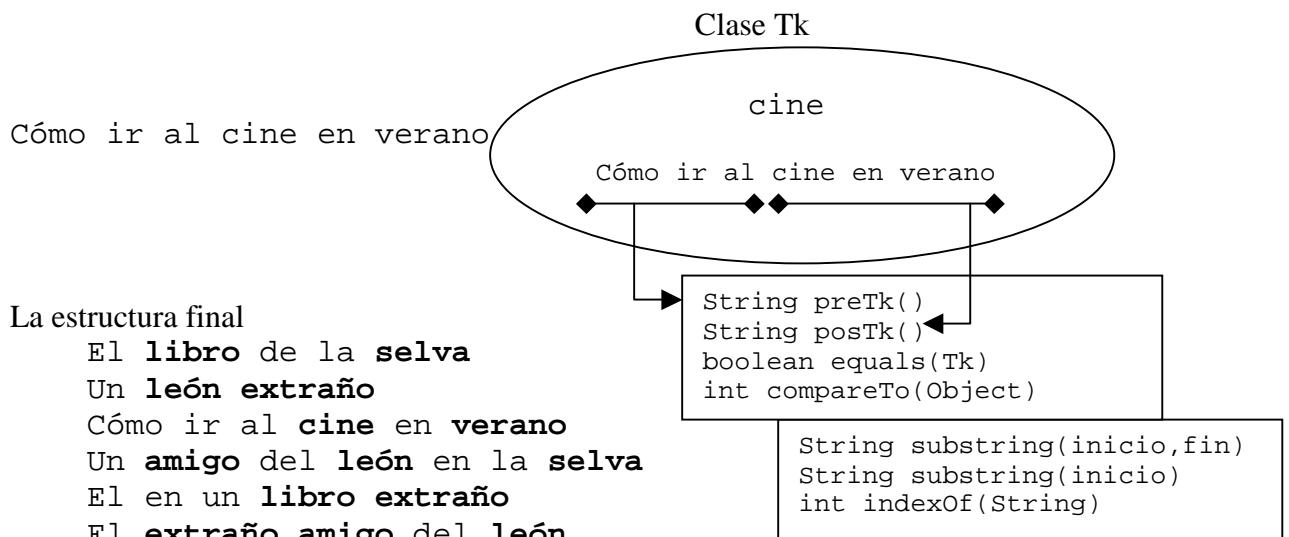
Donde cada título aparece tantas veces como palabras significativas tiene. Además, el título aparece partido por la palabra significativa y el principio y final de un título se separa con una barra vertical. Por ejemplo, “El extraño amigo del león” aparece por “amigo”, por “extraño” y por “león”. La lista resultante está ordenada por palabras clave (la primera palabra que aparece) y dentro de palabras claves por la frase completa.

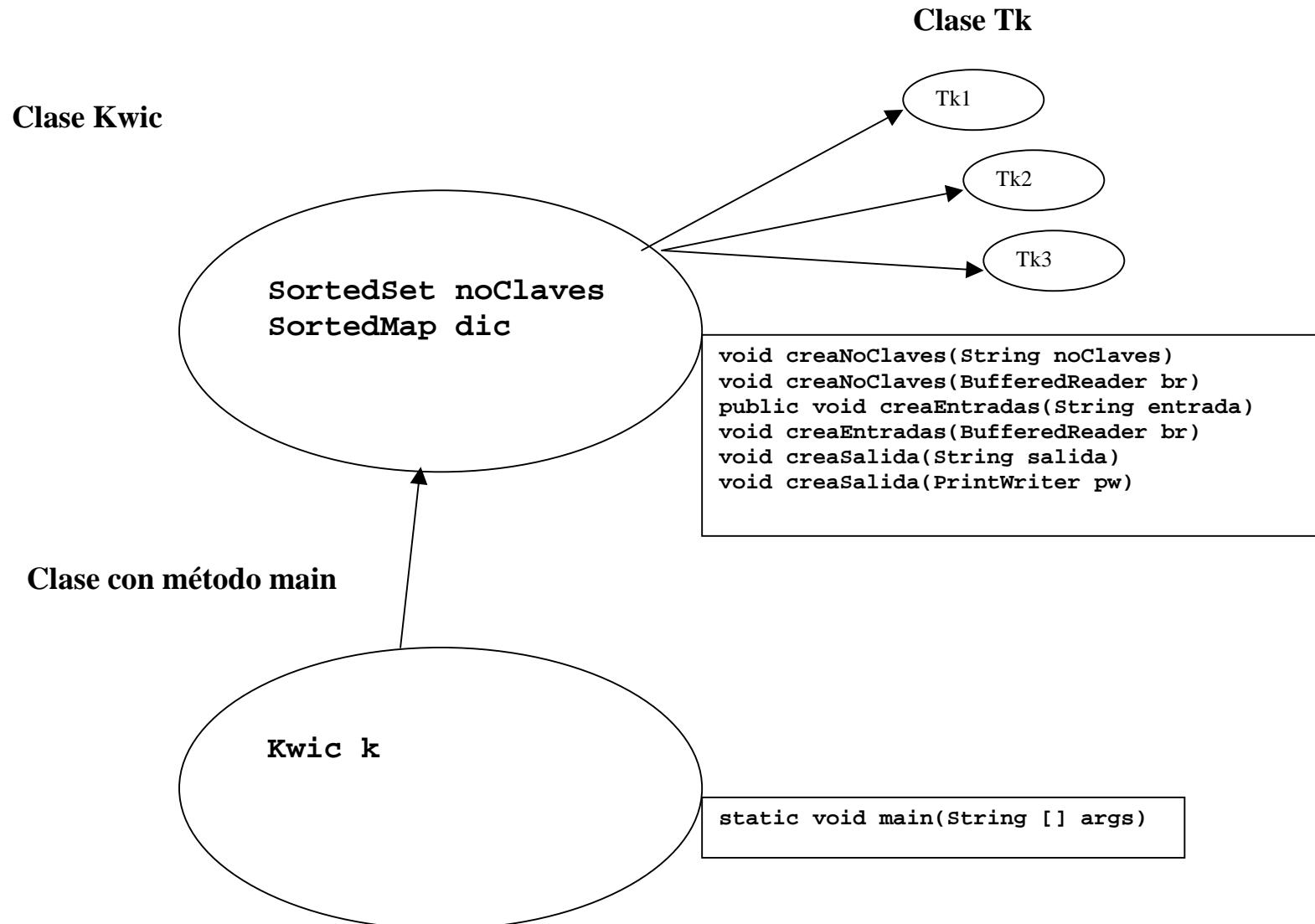
Conjunto de palabras no significativas

el la los las un una
de entre para por
y en desde hasta
sobre tras del
cómo al ir



Titulos KW





```
public class Tk implements Comparable {
    protected String palabra;
    protected String titulo;
    public Tk(String t, String p) {
        titulo = t;
        palabra = p;
    }
    public String preTk() {
        int i = titulo.indexOf(palabra);
        if (i<1)
            return "";
        return titulo.substring(0,i-1);
    }
    public String postTk() {
        int i = titulo.indexOf(palabra);
        if (i<0)
            return titulo;
        return titulo.substring(i);
    }
    public boolean equals(Tk tk) {
        if (titulo.equalsIgnoreCase(tk.titulo))
            return palabra.equalsIgnoreCase(tk.palabra);
        else
            return false;
    }
    public int compareTo(Object o) {
        Tk tk = (Tk)o;
        int i = palabra.compareToIgnoreCase(tk.palabra);
        if (i!=0)
            return i;
        else
            return titulo.compareToIgnoreCase(tk.titulo);
    }
}
```

```
import java.io.*;
import java.util.*;
public class Kwic {
    private Set snc;
    private SortedMap dic;
    public void creaNoClaves(String noClaves) {
        try {
            BufferedReader br= new BufferedReader(new FileReader(noClaves));
            creaNoClaves(br);
            br.close();
        } catch (IOException e) {
            System.out.println(e.getMessage());
            System.exit(1);
        }
    }
    public void creaNoClaves(BufferedReader br) throws IOException {
        ...
    }
    public void creaEntradas(String entrada) {
        ...
    }
    public void creaEntradas(BufferedReader br) throws IOException {
        ...
    }
    public void creaSalida(String salida) {
        ...
    }
    public void creaSalida(PrintWriter pw) {
        ...
    }
}
```