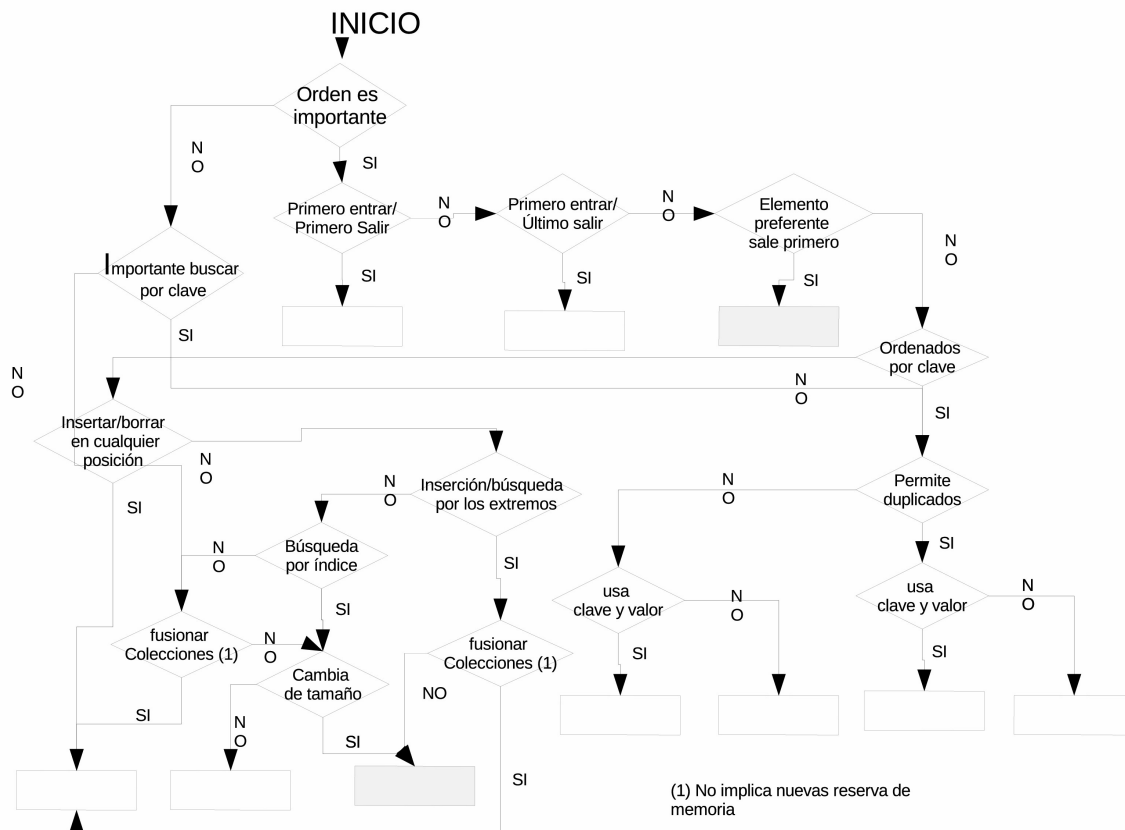


**Estructuras de Datos**  
**Curso 2014-2015. Convocatoria de Septiembre**  
**Grado en Ingeniería Informática.**  
**Doble Grado en Ingeniería Informática y Matemáticas**

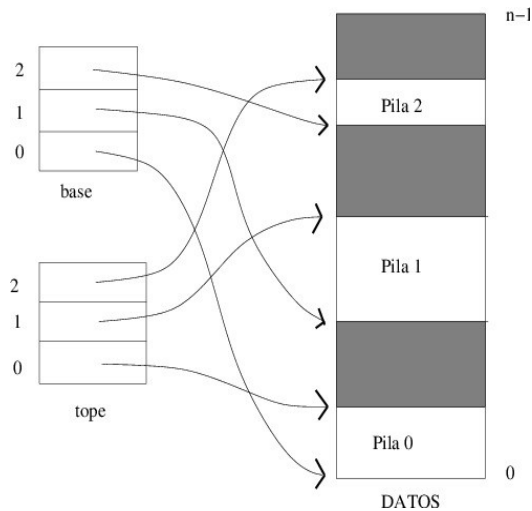
1.-

(a) (1 Punto) Completar la siguiente figura, indicando en los rectángulos la estructura de datos que utilizarías cuando un problema tiene las restricciones indicadas.



(b) (1 Punto) Razona tu decisión sobre los TDA que has ubicado en los recuadros grises en base a la representación interna del mismo así como considerando la eficiencia de las operaciones.

2.- (2 puntos) Se pretende almacenar  $k$  pilas en un único vector utilizando la estructura de datos sugerida en la figura siguiente (para el caso  $k=3$ )  
Implementa las funciones **push** y **pop** que insertan/borran un determinado entero en una de las pilas



3.- (2 puntos) Dada una lista de enteros con elementos repetidos, diseña (usando el TDA **list**) una función que construya a partir de ella una lista ordenada de listas, de forma que en la lista resultado los elementos iguales se agrupen en la misma sublista. La lista original no se ve modificada. Por ejemplo:

si **entrada**={1,3,4,5,6,3,2,1,4,5,5,1,1,7}, **salida**={{1,1,1},{2},{3,3},{4,4},{5,5,5},{6},{7}}

4.- (2 puntos) Sea T un árbol binario con n nodos cuyas etiquetas (de tipo string) son todas diferentes. Se define un k-nodo como un nodo **v** que cumple la condición de que el número de descendientes en el subárbol izquierdo de **v** difiere del número de descendientes del subárbol derecho al menos en k elementos. Usando el TDA bintree, diseña una función que tenga como entrada un árbol binario de string y como salida el listado de las etiquetas de los 5-nodos que posee

***list<string> cinconodos(bintree<string> & ab)***

5.- (2 puntos) Implementa un iterador que itere sobre las claves que sean números pares en una clase Diccionario definida como:

```
class Diccionario{
private:
    map<int, list<string> > datos;
    .....
    .....
}
```

Han de implementarse (aparte de las de la clase iterador) las funciones begin() y end().

**Tiempo: 3 horas**