

Práctico 1: Implementación de Operaciones

Licenciatura en Ciencias de la Computación - Profesorado en Ciencias de la Computación (plan viejo)

Ejercicio 1:

Dadas las siguientes estructuras:

- i. Lista secuencial desordenada (LSD).
- ii. Lista vinculada desordenada (LVD).
- iii. Lista vinculada ordenada (LVO).
- iv. Lista secuencial ordenada (LSO) (se usa búsqueda secuencial).

en las cuales se han almacenado, en el orden que aparecen, los elementos: 10, 25, 3, 38, 54, 5, 89, 17, 33, 41, 28, 71, 62, 39, 7, se pide:

- a) Mostrar cómo queda cada estructura con esos elementos almacenados, explicando cómo se fue transformando la estructura.
- b) Determinar si los siguientes objetos: 17, 22, 7, 1, 44, 99, y 50 forman parte del conjunto almacenado, utilizando las rutinas estudiadas y mostrando los pasos seguidos en cada caso.
- c) Las rutinas utilizadas en el punto **b)** arman la condición del `while` con un “*AND condicional*” (significa que se evalúa la *primer* condición y si es **falsa** *no* se continúa con la evaluación, de las siguientes) ¿Para qué sirve esta propiedad del *AND*?
- d) Modificar las rutinas utilizadas en el punto **b)**, reemplazando el operador *AND condicional* por sólo *AND*.

Ejercicio 2:

Se tiene un conjunto X almacenado en una lista vinculada. Según el pseudocódigo visto, cada iteración en la rutina `Pertenece` requiere de dos preguntas: una para $(p \ll \text{nil})$ y una para $(p \uparrow . \text{valor} \ll x)$. ¿Se podría omitir la pregunta $(p \ll \text{nil})$ en la condición de cada iteración? Justifique su respuesta y si es afirmativa explique cómo lo haría.

Ejercicio 3:

Se ha almacenado el conjunto X en una lista secuencial desordenada (LSD) y se quiere saber si un elemento pertenece a X o no.

- a) Resuelva, en pseudocódigo, utilizando la siguiente estrategia:
Tomar el elemento a buscar y almacenarlo como último elemento de la estructura. Realizar una búsqueda secuencial sobre la lista.
- b) ¿Qué diferencias encuentra con la estrategia utilizada en el **Ejercicio 1**?
- c) Si se quisiera usar esta misma técnica para una lista vinculada desordenada, ¿Sería útil hacerlo? ¿Cómo lo haría?

Ejercicio 4:

Se tiene un conjunto almacenado en:

- i. Lista secuencial ordenada en forma decreciente.
- ii. Lista vinculada ordenada en forma decreciente.

En cada caso, como último elemento, se agregó un “ $-\infty$ ” (objeto más pequeño que cualquier elemento posible del conjunto).

- a) Resolver, en pseudocódigo, la pertenencia de un elemento al conjunto, utilizando la estrategia de *examinación secuencial*.
- b) Analizar la utilidad de esta técnica en cada estructura.
- c) Suponga las listas ordenadas de menor a mayor. Analizar cómo se podría aplicar en ellas una técnica equivalente a la descripta.

Ejercicio 5:

Dado un conjunto almacenado en una Cola y dos elementos provistos por el usuario, se pide desarrollar un pseudocódigo que busque si el primer elemento provisto está en la estructura; si lo encuentra, éste debe ser reemplazado por el segundo elemento suministrado. Los elementos deberán quedar en el mismo orden en que fueron encontrados.

Ejercicio 6:

Dado un conjunto almacenado en una LSO, y dada como estrategia de examinación, la definida a continuación:

*Se examina una celda: si el elemento almacenado allí no es el buscado verificar si el elemento buscado es **mayor** que el contenido de dicha celda, en ese caso avanzar **dos** celdas; si es **menor** entonces retroceder **una** celda y finalizar la búsqueda preguntando si en esa celda se encuentra el elemento buscado. La búsqueda comienza en la primera celda de la lista.*

Se pide:

- a) Si el conjunto almacenado es { 6, 14, 22, 94, 39, 7, 52, 60, 82, 42 }. Decidir si los elementos 6, 40, 125 y 94 pertenecen a este conjunto, mostrando los pasos seguidos en cada caso.
- b) Si el conjunto almacenado es { 3, 32, 94, 39, 6, 52, 60, 88, 42 }. Decidir si los elementos 2, 40, 125 y 94 pertenecen a este conjunto, mostrando los pasos seguidos en cada caso.
- c) Desarrollar un pseudocódigo que resuelva este problema utilizando la estrategia detallada.

Ejercicio 7:

Dada una LSO con los siguientes valores almacenados en ella -8, 4, -5, 9, -12, 18, 25, 40, 60 decidir, utilizando *búsqueda binaria*, si los elementos 40, -10 y 84 pertenecen al conjunto. La consigna que se debe utilizar es: **trisección**, límite inferior inclusivo, límite superior inclusivo y segmento más grande a derecha. Mostrar paso a paso cómo se va analizando la estructura hasta decidir la pertenencia, o no, del elemento al conjunto.

Ejercicio 8:

Sabiendo que tiene información almacenada en cada una de las estructuras listadas a continuación, debe desarrollar, en pseudocódigo, los operadores que le permitan administrar cada una de las mismas

(alta, baja con confirmación, baja automática, evocación y modificación). Especificar, en cada caso, los parámetros que necesita cada rutina y las suposiciones que considere necesarias.

- a) Lista vinculada desordenada (LVD).
- b) Lista vinculada ordenada (LVO) en forma decreciente con finalización dada por contenido, (el equivalente a $-\infty$ en el dominio correspondiente).
- c) Lista secuencial desordenada (LSD).
- d) Lista secuencial ordenada (LSO) en forma creciente con *examinación secuencial* y finalización dada por contenido (el equivalente a $+\infty$ en el dominio correspondiente).

Ejercicio 9:

Se tiene el conjunto $X = \{22, 35, 64, 97, 52, 103, 61\}$ almacenado en un *Rebalse Abierto Lineal* (RAL) según la función $h : X \mapsto [0 \dots 10]$. La estructura que se obtuvo, insertando los elemento en el orden en que aparecen citados arriba en X , es la que sigue (el símbolo ★ indica que esa celda es virgen o nunca usada):

T:

0	97
1	★
2	★
3	22
4	35
5	52
6	61
7	★
8	★
9	103
10	64

$h(22) = 3$
$h(25) = 9$
$h(35) = 4$
$h(64) = 10$
$h(52) = 3$
$h(97) = 10$
$h(103) = 9$
$h(39) = 5$
$h(61) = 3$
$h(78) = 2$

Decidir si los elementos: 61, 78, 25, 97, 22 y 39 pertenecen o no al conjunto, mostrando los pasos seguidos para tomar una decisión.

Ejercicio 10:

Suponer a un conjunto almacenado en cada una de las estructuras listadas a continuación. Desarrollar, en pseudocódigo, el operador necesario para decidir si un elemento pertenece o no al conjunto en cada caso.

- i. Árbol Binario de Búsqueda (ABB) con celdas que contienen marca de fin.
- ii. Rebalse Separado (RS) con listas ordenadas.

Ejercicio 11:

Una variante de búsqueda binaria es la *bisección*. El proceso que sigue esta técnica es el siguiente:

Dividir sucesivamente la LSO en dos sublistas lo más parejas posible. Elegir una de ellas comparando el elemento buscado con un testigo en la lista. Repetir el proceso anterior en la sublista elegida hasta llegar a una lista de tamaño uno. Preguntar si el elemento buscado está allí.

- a) Desarrollar, en pseudocódigo, la estrategia detallada tomando como consigna: límite inferior inclusivo, límite superior exclusivo, **testigo** a derecha y segmento más grande a derecha.
- b) ¿Se le ocurre un caso en esta técnica pueda fallar? Si es así, ¿Cómo lo solucionaría?

Ejercicio 12:

Desarrollar, en pseudocódigo, el operador que le permita decidir la pertenencia de un elemento a un conjunto almacenado sobre una LSO usando búsqueda binaria. La consigna a utilizar será: **bisección**, límite inferior inclusivo, límite superior inclusivo, testigo a derecha y segmento más grande a izquierda.

Ejercicio 13:

Suponga un conjunto de enteros almacenados en una LVO. Desarrollar, en pseudocódigo, una función *ImprimeInverso* que imprima los elementos de la lista en *orden inverso*, a partir de una posición p ingresada por el usuario.

Ejercicio 14:

Suponga que ha almacenado un conjunto X en un árbol binario ordenado de búsqueda. Desarrolle en pseudocódigo las siguientes rutinas:

- a) *Mínimo (out x)*: permite recuperar el menor elemento de X ,
- b) *Máximo (out x)*: permite recuperar el mayor elemento de X .