

Algoritmos y Programación II (75.41) Cátedra Lic. Gustavo Carolo		Integrador 25/07/2005
Tema 1	Ejercicio	Puntaje
Apellido y Nombre		
Padrón		
Nota Final		
Corrigió		

### Práctica

- Dado el siguiente TDA PROVEEDORES compuesto por:
  - Arbol proveedores TDA AB (*ordenado por el campo proveedor*):
    - Proveedor (*clave ordenamiento*)
    - lista de productos: TDA LSC (lista simple)
      - Producto (*Clave de ordenamiento*)
      - Stock
      - Pcio vta.
  - Cola productos a cotizar
    - Producto
    - Cantidad solicitada

### Se pide:

-Definir todos los elementos de la estructura.  
 -Desarrollar un procedimiento abstracto **Buscar mejor cotizacion**, que recibe por parametro el **TDA PROVEEDORES**, un **rango de proveedores (desde y hasta)** y devuelve una lista cuyos elementos tengan proveedor, producto, precio (y este ordenada por proveedor y dentro del proveedor por producto) con los mejores precios de los productos de la cola. Los productos devueltos deberan poseer stock suficiente al pedido de cotizacion. Devolver una variable con el resultado de la operación o el tipo de error que crea conveniente.

**Nota:** Los procedimiento de búsqueda y/o actualización y/o inserción en el árbol deberán ser realizados en forma recursiva. Tener en cuenta que en las estructuras ordenadas **NO** debe recorrerse de mas si se busca por la clave, **en caso de recorrer de mas se considerara mal el ejercicio..**

2. Dado el TDA ABO (Arbol binario Ordenado), definir una nueva primitiva CANT\_ELEM\_EN\_RANGO que reciba el TDA\_ABO, un elemento desde, un elemento hasta, y devuelva la cantidad de elementos del arbol que esten en el rango de claves de los elementos dados. Los elementos desde y hasta pueden no estar en el arbol. Para comparar las claves **DEBE** usarse la funcion compara\_clave que recibe como parámetros dos elementos y de acuerdo al valor de sus claves devuelve -1 si es menor la clave del primer elemento, 0 si son iguales y 1 si es mayor la clave del primer elemento. Para optimizar la implementacion, **NO se pueden usar otras primitivas del tipo ni estructuras auxiliares. COMO EL ARBOL ESTA ORDENADO NO SE DEBE RECORRER DE MAS YA QUE SE RECORRE POR SU CLAVE.**

**Entregar la resolución de la Teoría y la Práctica en hojas separadas.**

### Condiciones para aprobar el Integrador:

- Deben tener ambas partes (teorica y practica) aprobadas:
- Para aprobar la parte practica deben estar hechos los dos ejercicios. Si alguno no esta hecho o esta Mal no se aprueba. Los ejercicios tienen que cumplir con lo pedido en el enunciado.

Algoritmos y Programación II (75.41) Cátedra Lic. Gustavo Carolo		Integrador 25/07/2005
Tema 2		Ejercicio
Apellido y Nombre		Puntaje
Padrón	Cuat. Cursada:	
Nota Final		
Corrigió		

### Práctica

1. Dado el siguiente TDA PRODUCTOS compuesto por:

- Arbol proveedores TDA AB (*ordenado por proveedor*)
  - Proveedor (*clave ordenamiento*)
  - lista de productos: TDA LSC (lista simple)
    - Producto (*Clave de ordenamiento*)
    - Stock
    - Pcio vta.
- Arbol productos internos TDA AB (*ordenado por producto*)
  - Producto (*clave de ordenamiento*)
  - Cantidad en stock
  - Ultimo Precio compra

#### Se pide:

- Definir todos los elementos de la estructura.

a) -Desarrollar un procedimiento abstracto **Comprar mejor precio**, que recibe por parametro el **TDA PRODUCTOS**, un **rango de proveedores (desde y hasta)**, un **producto y una cantidad a comprar** y actualice en el arbol de productos internos la cantidad en stock (la cantidad que tiene + la cantidad a comprar) y el precio ultimo compra con los datos del mejor precio de vta de los proveedores *comprendidos en el rango pasado por parametro* (considerar que debe satisfacer la cantidad a comprar). En caso de no existir el producto en el arbol de productos internos agregarlo. Devolver una variable con el resultado de la operación o el tipo de error que crea conveniente.

**Nota:** Los procedimiento de búsqueda y/o actualización y/o inserción en el árbol deberán ser realizados en forma recursiva. Tener en cuenta que en las estructuras ordenadas **NO** debe recorrerse de mas si se busca por la clave, **en caso de recorrer de mas se considerara mal el ejercicio.**

2. Dado el TDA ABO (Arbol binario Ordenado), definir una nueva primitiva CANT\_ELEM\_FUERA\_DE\_RANGO que reciba el TDA\_ABO, un elemento desde, un elemento hasta, y devuelva la cantidad de elementos del arbol que NO esten en el rango de claves de los elementos dados. Los elementos desde y hasta pueden no estar en el arbol. Para comparar las claves **DEBE** usarse la funcion compara\_clave que recibe como parámetros dos elementos y de acuerdo al valor de sus claves devuelve -1 si es menor la clave del primer elemento, 0 si son iguales y 1 si es mayor la clave del primer elemento. Para optimizar la implementacion, **NO se pueden usar otras primitivas del tipo ni estructuras auxiliares. COMO EL ARBOL ESTA ORDENADO NO SE DEBE RECORRER DE MAS YA QUE SE RECORRE POR SU CLAVE.**

**Entregar la resolución de la Teoría y la Práctica en hojas separadas.**

#### Condiciones para aprobar el Integrador:

- Deben tener ambas partes (teorica y practica) aprobadas:
- Para aprobar la parte practica deben estar hechos los dos ejercicios. Si alguno no esta hecho o esta Mal no se aprueba. Los ejercicios tienen que cumplir con lo pedido en el enunciado.