

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA
4ta. práctica (tipo B)
(Primer Semestre 2022)

Indicaciones generales:

- Duración: 2h 50 min.
- Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
- Los programas deben ser desarrollados en Ansi C. Si la implementación es diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
- El orden será parte de la evaluación.
- Se utilizarán herramientas para la detección de plagios, por tal motivo si se encuentran soluciones similares, se anulará la evaluación a todos los implicados y se procederá con las medidas disciplinarias dispuestas por la FCI.
- Para este laboratorio solo se permite el uso de las librerías **stdio.h**, **stdlib.h**, **string.h** y **math.h**
- Los archivos deben llevar como nombre su código de la siguiente forma **codigo_LabX_PY**
- Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.

Pregunta 1 (10 puntos)

La empresa desarrolladora de software MacroSoft, ha decidido lanzar al mercado su base de datos transaccional MacSQL 2023, por tal motivo está buscando los mejores algoritmos y estructuras de datos más eficientes, con el fin que su motor sea el más rápido y al fin vencer a Oráculo Database. Para esta tarea se le encomienda a Ud. que se encargue del manejo de la parte más importante de cualquier tabla, la validación de la llave primaria (PK), es así como se decide manejar las tablas de este nuevo motor con un ABB. A continuación, se muestra un ejemplo de cómo sería el árbol para una tabla con 3 campos:

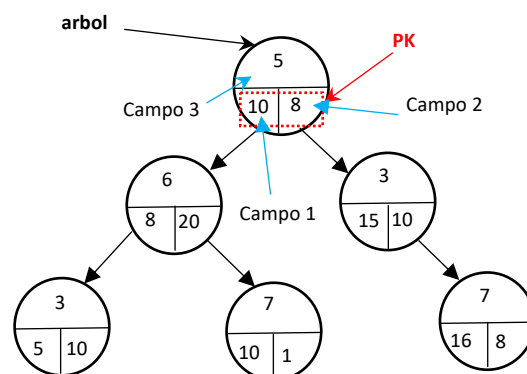


Imagen No. 1

El recorrido pre-orden de este árbol tabla será el siguiente:

(10 8 5) (8 20 6) (5 10 3) (10 1 7) (15 10 3) (16 8 7)

Como se observa en la imagen 1, cada registro o fila de la tabla se está representando como un nodo del ABB, además la llave primaria o PK está formada por los **campos 1 y 2**, lo que determina el orden en el ABB. A continuación, se le solicita un conjunto de operaciones a efectuarse en el árbol tabla, recuerde que todas las operaciones deben realizarse de la forma más eficiente posible, **evitando recorridos innecesarios o mover registros(nodos) que no tienen que ver con la transacción solicitada**, ya que se trata de tablas muy grandes y Microsoft requiere que su BD sea ágil:

- Desarrolle una función que **verifique** si se puede **insertar** un nuevo registro en la tabla, por ejemplo, si trata de insertar el registro (5 10 6), no es posible y deberá mostrar el mensaje "Cannot insert duplicate key", por ejemplo, si trata de insertar el registro (5 11 6) si es posible y deberá mostrar "Ok" (1 punto).
- Desarrolle una función que **verifique** si se puede actualizar los campos de la PK, por ejemplo, en el caso del registro (5 10 3) no es posible actualizar el campo1 con el valor de 15, por tal motivo deberá mostrar el mensaje "Violation of Primary Key constraint on update", si es posible debe mostrar "Ok". Recuerde que la validación es aplicable al campo1, campo2 o ambos (2 puntos).
- Desarrolle una función que luego de realizar la validación de la pregunta anterior **aplique** la actualización solicitada en los campos, en este caso si pueden actualizarse cualquiera de los 3 campos del registro (nodo). Para el caso de una actualización del campo 3 la complejidad debe ser lo más baja posible. No puede usar arreglos, matrices o TAD's auxiliares para esta pregunta, se le recomienda el uso de punteros tipo árbol y datos tipo elementoarbol (3 puntos).
- Desarrolle una función que **verifique** si es posible cambiar los campos de la PK, de los campos 1 y 2 por los campos 3 y 2. De acuerdo con el ejemplo de la imagen 1 esta tarea no es posible, pero si nuestro árbol tuviese los siguientes valores: (10 8 5) (8 20 6) (5 10 3) (10 1 7) (15 11 3) (16 8 7), si es posible llevar a cabo esta operación. En caso la operación no es posible deberá mostrar el siguiente mensaje "Violation of Primary Key constraint", si es posible debe mostrar "Ok". Para esta tarea puede usar solo pilas, colas o listas. Se le recomienda el uso de punteros y datos tipo elementoarbol (4 puntos).

Es importante para todas las preguntas, que al detectarse un registro que viole la PK, emita el resultado y así evite recorrer el árbol tabla, ya que se sabe que cuenta con un millón de registros. Como datos de entrada puede usar los valores que se muestran en el ejemplo, así como los valores que aparecen en las diferentes operaciones, por tal motivo los campos siempre serán numéricos y con fines de desarrollo siempre serán 3 campos con una PK formada por 2 campos.

Pregunta 2 (10 puntos)

En la gestión de productos alimenticios de un mismo tipo que entran y salen de un almacén se suele aplicar la regla de que primero se deben despachar los productos cuya fecha de caducidad son las que están más próximas a la fecha de despacho. De acuerdo con las recomendaciones de un egresado de Algoritmia, se ha implementado un árbol de búsqueda binaria para almacenar los lotes de productos de un mismo tipo de producto alimenticio, considerando que cada lote tiene una fecha de caducidad diferente a la de cualquier otro lote.

Los datos de un lote se conforman de su fecha de vencimiento (p.e. 20220815) y de la cantidad de productos que corresponden a dicho lote.

Si por ejemplo, el registro de los lotes de productos alimenticios se da en el siguiente orden:

(20220815 - 40), (20220710 - 50), (20220720 - 50), (20220630 - 50), (20220624 - 40), (20220615 - 40), (20220618 - 40), (20220930 - 30)

El árbol formado sería como se muestra en la Imagen 2.

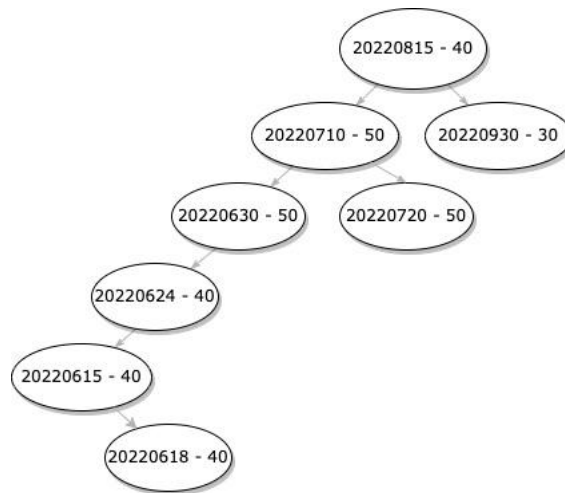


Imagen 2. ABB que guarda los lotes de un mismo tipo de productos ordenados por la fecha de caducidad.

Cada vez que se realizan solicitudes de unidades de productos, se procede a buscar el lote con la menor fecha de caducidad y se le reduce la cantidad de productos solicitados. En el caso que la cantidad de productos solicitados coincida con la cantidad disponible en el lote de menor fecha de caducidad, entonces se procede a eliminar el nodo de dicho lote. Si la cantidad de productos solicitados excede a la cantidad disponible en el lote de menor fecha de caducidad, entonces se elimina el nodo de dicho lote, se reduce la cantidad de productos solicitados en la cantidad de productos del lote eliminado y se procede a buscar el siguiente lote con menor fecha de caducidad para repetir el proceso hasta lograr que la cantidad de productos solicitados sea igual a 0. Si la cantidad de productos solicitados excede la cantidad disponible en todo el árbol, se despachan todos los productos disponibles y se muestra que no se ha logrado despachar la diferencia entre lo solicitado y lo disponible.

Durante el día se pueden realizar varias solicitudes de despacho. Si en el ejemplo del árbol formado en la Imagen 2, se realizan tres solicitudes de despacho, la primera de 20 unidades, la segunda de 40 unidades y la última de 200 unidades, la ejecución del programa sería:

La representación, en preorden, del árbol de lotes de productos es:

(20220815 - 40), (20220710 - 50), (20220630 - 50), (20220624 - 40), (20220615 - 40), (20220618 - 40), (20220720 - 50), (20220930 - 30)

Ingrese la cantidad de solicitudes de despacho: 3

Ingrese la cantidad de unidades para el despacho 1: 20

La representación, en preorden, del árbol de lotes de productos luego del despacho 1 es:

(20220815 - 40), (20220710 - 50), (20220630 - 50), (20220624 - 40), (20220615 - 20), (20220618 - 40), (20220720 - 50), (20220930 - 30)

Ingrese la cantidad de unidades para el despacho 2: 40

La representación, en preorden, del árbol de lotes de productos luego del despacho 2 es:

(20220815 - 40), (20220710 - 50), (20220630 - 50), (20220624 - 40), (20220618 - 20), (20220720 - 50), (20220930 - 30)

Ingrese la cantidad de unidades para el despacho 3: 200

La representación, en preorden, del árbol de lotes de productos luego del despacho 3 es:

(20220815 - 40), (20220720 - 10), (20220930 - 30)

Se le solicita implementar lo siguiente:

- a) Desarrolle las estructuras y funciones necesarias para el ingreso adecuado de los datos en el árbol (1.0 punto).
- b) Desarrolle una función que realice la actualización del árbol cuando se realiza un despacho. Para el desarrollo puede apoyarse en cualquier otro TAD, menos arreglos (7.0 puntos).
- c) Desarrolle la función principal que, luego del ingreso de lotes de productos, realice la representación del árbol inicial, solicite la cantidad de solicitudes de despacho, por cada solicitud se ingrese la cantidad de productos a despachar y se muestre la representación en preorden del árbol luego de cada despacho (2.0 puntos).

Profesores del curso: Juan Manuel Chau
 Rony Cueva
 Johan Baldeón

San Miguel, 10 de junio del 2022