

IBDD 2009 - Primer Parcial

Segunda Fecha -- 12/09/2009

Tema II

1. Una fábrica debe procesar mensualmente 3 (tres) archivos detalles de horas extras para totalizar la liquidación de sueldos de sus empleados. Los registros de los tres archivos son iguales y se componen del número de empleado, fecha y cantidad de horas extras. Además, la fábrica cuenta con otro archivo detalle extra donde para cada empleado registra: número de empleado, sueldo básico y el monto que cobra cada hora extra. Todos los archivos están ordenados por número de empleado y en los tres detalles de horas extras puede haber 0, 1 o más registros de cada empleado. Desarrolle un algoritmo (programa) para producir un nuevo archivo (ordenado por el mismo criterio) con la siguiente estructura: número de empleado y total a cobrar (liquidación total del sueldo). A medida que genera este nuevo archivo debe tener en cuenta que:
 - Si el básico del empleado supera los \$2500 debe realizarle un descuento del 3% a la liquidación total del sueldo. Si el básico no supera los \$2500 debe informar en un archivo de texto (además de grabarlo en el nuevo archivo) el número de empleado, el total de horas extras realizadas y el monto total correspondiente a la liquidación de su sueldo.

NOTA: Los archivos deben ser recorridos una sola vez.

2. Dado el árbol B+ que se muestra abajo, con capacidades en nodos internos y hoja para 3 y 4 elementos respectivamente, mostrar los estados sucesivos completos al efectuarle las siguientes operaciones en el orden en que se especifican y respetando la convención de dejar más cargados a los nodos izquierdos cuando haya particiones y de priorizar hermanos derechos cuando haya que intentar balanceos: alta 100, baja 898 y baja 673. Justificar cada operación.

NODO 0 (i, 3 elemento/s)

2(406) 4(632) 3(777) 1

NODO 2 (h, 4 elemento/s)

(151)(161)(562)(371) Siguiente hoja: 4.

NODO 4 (a, 2 elemento/s)

(406)(523) Siguiente hoja: 3

NODO 3 (h, 3 elemento/s)

(632)(673)(761) Siguiente hoja: 1

NODO 1 (h, 2 elemento/s)

(777)(898) Siguiente hoja: 0

3. Dado el Archivo Directo con Dispersión Extensible-Modular (Bits Sufijos) y Bloques con Capacidad para 3 Registros que se muestra abajo, mostrar los estados sucesivos completos al efectuarle las siguientes operaciones en el orden en que se especifican: alta 2811, baja 25614, alta 5176. Justificar cada operación.

Tabla: $\begin{matrix} 4 & 3 & 1 & 0 & 4 & 2 & 1 & 0 \end{matrix}$

Bloques Libres: -

Claves de elementos en Bloques:

Bloque 0, id 4: 31503 28499 9855

Bloque 1, id 4: 25614

Bloque 2, id 8: 9125 42813 46173

Bloque 3, id 8: 8929 37001

Bloque 4, id 4: 9608 23359 62536

Luciano

IBDD 2010 - Primer Parcial

Primer Fecha – 26/06/2010

Tema II

1. Dada la siguiente estructura perteneciente a una organización de archivo de registros de longitud variable:

```
tPersona = Record
    dni: LongInt;
    nombre: String;
end;

tBloque = Record
    cantRegs: Word; {cantidad actual de registros en el bloque}
    contenido: Array[1..CapacBloque] of Byte; {CapacBloque es una constante con un valor predefinido}
end;

tPersonas = Record
    arch: File of tBloque; {archivo de personas}
    espLibre: File of Word; {archivo de control de bytes libres por bloque del archivo de personas}
    bloque: tBloque; {último bloque leído de personas}
    iBloque: Word; {índice de posición dentro del bloque}
    espLibreBloque: Word;
end;
```

Implemente las siguientes primitivas:

Procedure Primera(var pp: tPersonas; var p: tPersona); {Devuelve la primer persona del archivo.}

Procedure Siguiente(var pp: tPersonas; var p: tPersona); {Devuelve la siguiente persona del archivo o p.dni=0 en caso de que no hubiera más personas.}

2. Dado un árbol B+ con capacidades de 3 y 4 elementos en nodos internos y hoja respectivamente, muestre los estados sucesivos al realizar la siguiente secuencia de operaciones: *baja 193, alta 815 y baja 459*. Siempre que haya que balancear, agregar o eliminar nodos debe trabajarse con hermanos derechos, y ante disparidad de carga en particiones y balanceos debe quedar más cargado el hermano izquierdo.

NODO 0: (i, 1 elem.) 7 (740) 6

NODO 7: (i, 1 elem.) 2(459) 4

NODO 2: (h, 2 elem.) (52)(193) Sig. hoja: 4

NODO 4: (h, 3 elem.) (459)(471)(672) Sig. hoja: 1

NODO 6: (i, 1 elem.) 1(900) 5

NODO 1: (h, 4 elem.) (740)(791)(800)(836) Sig. hoja: 5

NODO 5: (h, 2 elem.) (900)(950) Sig. hoja: 0

Nota: Para cada paso justificar y dibujar el árbol resultante de manera completa.

3. Dado un archivo directo con cubetas para dos registros y dispersión lineal (saturación progresiva) con función de dispersión mod 9 y con el estado inicial de abajo, muestre su estado luego de cada operación, informando cuáles cubetas leyó y cuáles escribió para efectuar la: *alta 661, alta 777, baja 741*.

Desbordado

0	N		
1	N	523	694
2	N	668	
3	N	741	246
4	N	814	229
5	N	734	
6	N		
7	N		
8	N	215	

Nota: Para cada paso justificar y dibujar la tabla resultante de manera completa.

1. Se posee un archivo que tiene información de un censo a nivel mundial. Cada registro contiene la siguiente información: Continente, País, Ciudad, Cantidad de Varones y Cantidad de Mujeres. El archivo está ordenado por Continente, País. Realice un módulo que reciba y abra el archivo correspondiente y liste en un archivo de texto la siguiente información:

Continente:

País:

Ciudad Cantidad de Varones Cantidad de Mujeres

Total de habitantes País:

Porcentaje de mujeres del País:

País:

Total Habitantes Continente:

Continente:

Nota: Considere que no hay ciudades repetidas.

2. Dado un árbol B con capacidades de 3 y 5 elementos en nodos internos y hoja respectivamente, muestre los estados sucesivos al realizar la siguiente secuencia de operaciones: *alta 700, baja 623, baja 817, alta 900 y alta 750*. Siempre que haya que balancear, agregar o eliminar nodos priorizar hermanos derechos, y ante disparidad de carga en particiones y balanceos debe quedar más cargado el hermano izquierdo.

NODO 0 (i, 3 elemento/s)

1(324) 4(433) 3(623) 2

NODO 1 (h, 3 elemento/s)

(7)(97)(208)

NODO 4 (h, 2 elemento/s)

(338)(399)

NODO 3 (h, 2 elemento/s)

(474)(508)

NODO 2 (h, 5 elemento/s)

(676)(694)(806)(817)(887)

3. Dado un archivo directo con cubetas para tres registros y dispersión con área de desborde separada encadenada (el área de desborde posee capacidad para un sólo registro) con función de dispersión mod 9 y con el estado inicial de abajo, muestre su estado luego de cada operación, informando cuáles cubetas leyó y cuáles escribió para efectuar la: *alta 156, alta 705, alta 107*.

Bloque Desborde Contador

0	-1	3	459	270	954
1	-1	2	946	973	
2	-1	1	731		
3	-1	3	741	426	462
4	-1	0			
5	-1	2	446	410	
6	-1	2	123	789	
7	-1	2	376	484	
8	-1	3	458	305	809

IBBDD 2008 - Primer Parcial - PLAN 90

Segunda Fecha - 16/08/2008

Tema II

1. Realice un programa para crear y cargar un archivo de registros de longitud variable con la información del personal administrativo de la facultad de informática, esta información se lee desde teclado hasta que llegue la persona con DNI 0. De cada persona se lee: DNI, Apellido y Nombre. El Apellido y el Nombre de la persona es de Longitud Variable.

Una vez generado el archivo, realice un modulo que reciba el DNI de una persona (que seguro existe) y lo borre lógicamente del archivo generado.

2. Dado el árbol B+ de abajo, que puede tener hasta 6 claves en las hojas como máximo (y 3 como mínimo) y hasta 4 claves en los nodos internos como máximo (y 2 como mínimo):

A - Dibuje cómo queda el árbol resultante para la baja de la clave 30474.

B - Dibuje cómo queda el árbol resultante del punto A para el alta de la clave 50000.

0: 8(30549) 7
8: 2(10463) 5(22905) 3 7: 6(41577) 1(57336) 4(61038) 9
2: (801)(1605)(4882)(5893)(7397)(9108) 5 6: (30549)(31815)(38385) 1
5: (10463)(12746)(16290) 3 1: (41577)(43074)(47485)(51304)(51376)(56752) 4
3: (22905)(24213)(30474) 6 4: (57336)(57808)(59332)(59570) 9
9: (61038)(61072)(63398) 0

NOTA: Justifique y explique como obtiene el resultado en cada operación realizada sobre el árbol.

3. Se tiene que distribuir un archivo y se decide utilizar el método de hashing extensible (dispersión dinámica), se sabe que cada cubeta tiene una capacidad para 2 registros, se pide mostrar como se genera el archivo si las claves y el resultado de la función de hash son las siguientes:

1	Makemake	0000 1010	2	Caronte	0101 0001
3	Santa	1010 1100	4	Sedna	0001 1110
5	Quaoar	0110 0011	6	Orcus	1100 0110
7	Ixión	1110 0101	8	Varuna	0011 1001

NOTA: Justifique y explique como obtiene el resultado en cada operación realizada.

IBBDD 2008 - Primer Parcial - PLAN 2003

Tercer Fecha - 13/09/2008

Tema I

1. Se debe crear un archivo (con registros de longitud fija) de asociados a una tarjeta de crédito, de cada asociado se registra: número de asociado, dni, apellido, nombre, domicilio, fecha de nacimiento y fecha de asociación. Escriba un programa para crear y cargar dicho archivo leyendo los datos de cada asociado desde teclado. La lectura finaliza cuando llega el asociado con número 0 (cero). La creación del archivo deberá ser realizada con reaprovechamiento de espacio libre controlado por una LISTA INVERTIDA dentro del mismo archivo.

Una vez leída y cargada toda la información, escriba un modulo que reciba el archivo anteriormente cargado y los datos de un socio previamente leído y realice el alta del mismo con reaprovechamiento de espacio libre controlado por la lista invertida dentro del mismo archivo. Tenga en cuenta que entre la carga y la invocación al modulo se pueden haber borrado algunos registros del archivo.

2. Dado el árbol B+ de abajo, que puede tener hasta 4 claves en las hojas como máximo (y 2 como mínimo) y hasta 3 claves en los nodos internos como máximo (y 1 como mínimo):

A - Dibuje cómo queda el árbol resultante para el alta de la clave 125.

B - Dibuje cómo queda el árbol resultante del punto A para la baja de la clave 937.

C - Dibuje cómo queda el árbol resultante del punto B para la baja de la clave 595.

NODO 0 (i, 3 elemento/s) 2 (sucesor izquierdo del primer elemento) (504) 4(685) 3(937) 1

NODO 2 (h, 4 elemento/s)

(69)(192)(262)(442) Siguiendo hoja: 4

NODO 4 (h, 2 elemento/s)

(504)(595) Siguiendo hoja: 3

NODO 3 (h, 2 elemento/s)

(685)(883) Siguiendo hoja: 1

NODO 1 (h, 2 elemento/s)

(937)(952) Siguiendo hoja: 0

NOTA: Justifique y explique brevemente como obtiene el resultado en cada operación realizada sobre el árbol.

3. Grafique la evolución de un archivo directo organizado en bloques con capacidad para dos registros y dispersión extensible con el método del resto (modulo tamaño de la tabla), realizando las altas en el orden que se indica a continuación:

1134, 2648, 946, 2215, 5214, 3801, 5928, 6593, 4626, 5849

NOTA: Justifique y explique brevemente como obtiene el resultado en cada operación realizada.

1. Dada la siguiente estructura perteneciente a una organización de archivo de registros de longitud variable:

```
tAlumno = Record
  legajo : LongInt;
  nom_ape: String;
end;

tBloque = Record
  cantRegs: Word; {cantidad actual de registros en el bloque}
  contenido: Array[1..CapacBloque] of Byte; {CapacBloque es una constante con un
valor predefinido}
end;

tAlumnos = Record
  arch: File of tBloque; {archivo de alumnos}
  espLibre: File of Word; {archivo de control de bytes libres por bloque del archivo
de alumnos}
  bloque: tBloque; {último bloque leído de alumnos}
  iBloque: Word; {índice de posición dentro del bloque}
  espLibreBloque: Word;
end;
```

Implemente la siguiente primitiva:

Procedure Agregar (var pp: tAlumnos; p: tAlumno); {Agrega un alumno en el primer bloque que tenga espacio o, si no hay ninguno, en un bloque nuevo al final del archivo, sin validar unicidad de registros.}

2. Dado el árbol B+ que se detalla más abajo, con orden 4, es decir, capacidad para 3 claves, muestre los estados sucesivos al realizar la siguiente secuencia de operaciones: +15, +55, -45, -33. **Justifique brevemente cada operación.**

Nodo 2: 3, i, 0(47)1(63)4(70)3

Nodo 0: 3, h, (23)(33)(45) -> 1

Nodo 1: 3, h, (47)(52)(59) -> 4

Nodo 4: 3, h, (63)(65)(69) -> 3

Nodo 3: 3, h, (70)(79)(90) -> -1

3. Para las claves siguientes, realice el proceso de dispersión mediante el método de hashing extensible, sabiendo que cada nodo tiene capacidad para 2 (dos) registros. El número natural indica el orden de llegada de las claves.

1	Buenos Aires1001	2	San Juan0100
3	Entre Ríos1110	4	Corrientes0010
5	San Luis0101	6	Tucumán0111
7	Río Negro0011	8	Jujuy1111