U N I V E R S I D A D   D E   C H I L E
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION (DCC)
AV. BLANCO ENCALADA 2120, CASILLA 2777, STGO. CHILE; TEL:+56(2)6892736, FAX:+56(2)6895531

CC50C - Estructuras de Datos Avanzadas - Profesor: Ricardo Baeza Yates
Tarea 2: Estructuras de Datos en Disco
Plazo: Día del examen

1)

a) Encuentre el orden m óptimo de un árbol B para minimizar el tiempo necesario para encontrar en que bucket está un elemento (es decir, leyendo el camino completo de la raíz hasta el último nivel). Puede considerar que la altura promedio de una árbol B de N buckets es $\log_{m/\ln 2} N$, que cada nodo del árbol son $2m+1$ punteros y $2m$ números enteros, que el tiempo de seek en un acceso a disco es 20 milisegundos y que el tiempo de transferencia es de 10 Megabytes por segundo. Para procesar el nodo, considere búsqueda binaria y búsqueda secuencial, suponiendo que cada comparación requiere 0.1 milisegundos incluyendo otras instrucciones. En base al resultado indique cual es la mejor opción.

b) Calcule la fracción que ocupa el índice de un árbol B^+ usando la estructura del formato anterior, en función del número de records N, el tamaño del record en palabras de 4 bytes T, el orden del árbol m y el tamaño del bucket b (en número de records). Considere el peor caso.
Haga el mismo cálculo para Desorden Acotado, usando los mismos parámetros para el índice y una tabla de hashing de tamaño $M+1$, incluyendo el bucket de overflow. Compare ambos casos usando $N = 10$ millones, $b = 10$, $m = 10$, $T = 10$ y $M = 10$. Comente si alguno de estos índices cabe en memoria principal.

2) Escriba un simulador de Desorden Acotado, en el lenguaje de programación de su preferencia, usando todas las suposiciones planteadas en clase (elementos aleatorios, función de hashing aleatoria uniforme) y encuentre el tamaño óptimo del bucket de overflow (que en este caso será variable) que maximiza la utilización de espacio en los buckets.
Recuerde que la probabilidad de insertar un elemento en un multibucket depende sólo del número de elementos que el tiene y por lo tanto bastará con almacenar el número de elementos por bucket. Esto mismo es suficiente para la manipulación de la tabla de hashing.
Encuentre los valores óptimos para $b = 10$ y M variable y viceversa, planteando, si es posible, un modelo para el óptimo. Luego grafique el óptimo para el caso en que ambos parámetros varían.

Nota: por cada día hábil de atraso o fin de semana se descontará medio punto.