CS 351 DATA ORGANIZATION & MANAGEMENT

FALL 2010

QUIZ 4/ SECTION-1 *(Date given: December 2, 2010)*

Bank Account

n (number of records) = 100,000

R (record size) = 400 bytes

B (block size) = 2400 bytes

r (rotational latency time) = 0.5 ms

s (seek time) = 9 ms

btt (block transfer time) = 0.5 ms

ebt (effective block transfer time) = 0.6 ms

key (key size) = 8 bytes

link (link size) = 4 bytes

fullness rate = 70%

Indexed File (B+ Tree)

a. What is the file size in terms of data nodes?

b. What is the expected number of index nodes at the level before data nodes?

c. How much time is needed to find average account balance?

d. How much time is needed to find average account balance if the data nodes are not connected to each other?

Solution:

a. File size in terms of data nodes is calculated as follows:

File size = (number of records * record size) / (Block size * fullness rate)

= (100,000 * 400 bytes) / (2400 bytes * 0.70)

= 23908.52 data nodes

≅ 23909 data nodes

b. First, we need to find expected number of pointers per index node.

Expected number of pointers/index node = (Block size * fullness rate) / (key size + link size)

= (2400 bytes * 0.70) / (8 bytes + 4 bytes)

= 140 pointers/index node

Now, we can find the number of index nodes in the last layer of index file:

→ Number of data nodes / Expected number of pointers/index node

= 23909 / 140 = 170.778 ≅ 171 index nodes

c. To get the account of the first data node, we make two disk accesses. For the other data nodes, we make only one disk access for each data node. Therefore, the time needed to find the average account balance is calculated as follows:

t = 2(s + r + btt) + (Number of data nodes – 1) * (s + r + btt)

= (Number of data nodes + 1) * (s + r + btt)

= 23910 * 10 ms = 239100 ms

(We accept also the following t: t = number of data nodes * (s + r + btt))

d. If data nodes are not connected to each other, we need to make two disk accesses for each data node. So the time needed to find the average account balance is:

t^{l} = 2 * Number of data nodes * (s + r + btt)

= 2 * 23909 * 10ms = 478180 ms

QUIZ 4/ SECTION-2 *(Date given: December 2, 2010)*

Student

n (number of records) = 200,000

R (record size) = 600 bytes

B (block size) = 2400 bytes

r (rotational latency time) = 2 ms

s (seek time) = 10 ms

btt (block transfer time) = 0.5 ms

ebt (effective block transfer time) = 0.6 ms

key (key size) = 8 bytes

link (link size) = 4 bytes

fullness rate = 70%

Indexed File (B+ Tree)

a. What is the file size in terms of data nodes?

b. What is the expected number of index nodes at the level before data nodes?

c. How much time is needed to find a single record?

d. How much time is needed to find average GPA?

e. How much time is needed to find average GPA if the data nodes are not connected to each other?

Solution:

a. File size in terms of data nodes is calculated as follows:

File size = (number of records * record size) / (Block size * fullness rate)

= (200,000 * 600 bytes) / (2400 bytes * 0.70)

= 71428.57 ≅ 71429 data nodes

b. First, we need to find expected number of pointers per index node.

Expected number of pointers/index node = (Block size * fullness rate) / (key size + link size)

= (2400 bytes * 0.70) / (8 bytes + 4 bytes)

= 140 pointers/index node

Now, we can find the number of index nodes in the last layer of index file:

→ Number of data nodes / Expected number of pointers/index node

= 71429 / 140 = 510.207 ≅ 510 index nodes

c. The time needed to find a single record is computed as follows:

T_F = 2 * (s + r + btt) = 2 * 12.5 = 25 ms

d. To get the GPA of the first data node, we make two disk accesses. For the other data nodes, we make only one disk access for each data node. Therefore, the time needed to find the average GPA is calculated as follows:

t = 2(s + r + btt) + (Number of data nodes – 1) * (s + r + btt)

= (Number of data nodes + 1) * (s + r + btt)

= 71430 * 12.5 = 892875 ms

(We accept also the following t: t = number of data nodes * (s + r + btt))

e. If data nodes are not connected to each other, we need to make two disk accesses for each data node. So the time needed to find the average account balance is:

t^1 = 2 * Number of data nodes * (s + r + btt)

= 2 * 71429 * 12.5 = 1785725 ms

QUIZ 4/ SECTION-3 *(Date given: December 3, 2010)*

Bank Account

Bk (number of data nodes) = 1000

s (seek time) = 10 ms

r (rotational latency time) = 2 ms

btt (block transfer time) = 1 ms

ebt (effective block transfer time) = 0.5 ms

fullness rate = 70%

Consider a B+ file

- Each data node can contain max 14 records.
- Data nodes are connected to each other.

a. How much time is needed to find a single record?

b. How much time is needed to find the next record?

c. How much time is needed to find average account balance?

d. How much time is needed to find average account balance if the data nodes are not connected to each other?

Solution:

a. The time needed to find a single record is computed as follows:

$T_F = 2 * (s + r + btt) = 2 * 13 = 26$ ms

b. First, we need to find expected number of records in a data node.

$\rightarrow 14 * 0.7 = 9.8 \cong 10$ records/data node

If the next record we want to find is in the same data block with the previous record fetched, it requires no cost. However, if the next record is in another data node, then we need to make a disk access. So the time needed to find the next record is computed as follows:

$T_N = (s + r + btt) / 10 = 13 / 10 = 1.3$ ms

c. To get the account of the first data node, we make two disk accesses. For the other data nodes, we make only one disk access for each data node. Therefore, the time needed to find the average account balance is calculated as follows:

t = 2* (s + r + btt) + (Number of data nodes – 1) * (s + r + btt)

= (Number of data nodes + 1) * (s + r + btt)

= 1001 * 13 ms = 13013 ms

(We accept also the following t: t = 1000 * (s + r + btt))

d. If data nodes are not connected to each other, we need to make two disk accesses for each data node. So the time needed to find the average account balance is:

$t^{'}$ = 2 * Number of data nodes * (s + r + btt)

= 2 * 1000 * 13 = 26000 ms