SOLUTION

INFSCI1022: Database Management Systems,

Spring 2012

HW 5: Storage and Indexing [100 pts]

You are running a DBMS on a file system with 3 kByte disk block size.

Reminder: 1kByte=1024bytes, 1 MByte = 1024 kBytes.

Q1 [15 pts]

The size of each tuple is 133 Bytes. There are 100,000 tuples in table T. Column Id stores sequential tuple numbers starting from 1 (e.g. : 1,2,3,4,5,...). The last value of Id is 100 000. How much of disk space you will need to store this table? Show your calculations.

Your answer (just one number in Mbytes): ≈12.7382 Mbytes
Your calculations:

All information will be stored in blocks. Tuples have to fit completely in block.
Thus, floor(3*1024/133 = 23.0977) = 23 complete tuples can be stored in 1 block.
ceiling(100,000 / 23 = 4,347.826) = 4,348 blocks are required to store all tuples. 3*4,348 = 13,044 kBytes ≈12.7382 Mbytes

Q2 [15 pts]

Assume that table T defined in question 1 has a dense primary index on column Id. The size of the index record is 2 bytes + 3 bytes for the pointer. How much disk space you will need to store the information of the table T (including both data and index)? Show your calculations.

Your answer (just one number in Mbytes): 13.2157Mbytes

Your calculations:

Dense index requires 1 index record per search key value. The size of each indexing record is 5 bytes (2 + 3). In 1 block can fit floor(3*1024/5 = 614.4) = 614 records.

Thus we need ceiling(100,000/614 = 162.866) =163 blocks to store all indexes records, which will need 3*163 = 489 Kbytes ≈ 0.4775 Mbytes on disk.

Totally we need 12.7382 (from Q1, for data) + 0.4775≈ 13.2157Mbytes on disk

Q3 [15 pts]

Assume that table T is defined in the same way as in question **1**. You execute a query: "select * from T where Id=54321". How many blocks and how many bytes will be read from the disk? Show your calculations.

Your answer (just two numbers: # of blocks; # of Mbytes): 4,348 blocks; ≈12.7382 Mbytes

Your calculations:

Without indexing DBMS will have to read the whole table, thus 4,348 blocks (from question 1) which equal to ≈12.7382 Mbytes

Q4 [15 pts]

Assume that table T is defined in the same way as in question **2**. You execute a query: "select * from T where Id=54321". How many blocks and how many bytes will be read from the disk? Show your calculations.

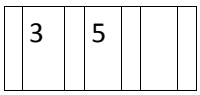
Your answer (just two numbers: # of blocks; # of Mbytes): 90 blocks; 0.2637 Mbytes.

Your calculations:

From **Q2** we know that 614 indexing records fit into one block. Thus the information about tuple with Id = 54321 is located in the 89 index block (celling(54321 / 614) = 89). In total DBMS will read 89 index blocks + 1 data block = 90 * 3 = 270 kBytes ≈ 0.2637 Mbytes.

Q5 [10 pts]

Is this a valid B+ tree?



Yes. That is the root, no condition applied. See slide 13 of pdf slides on Indexing lecture. In special cases, last point.

Q6 [15 pts]

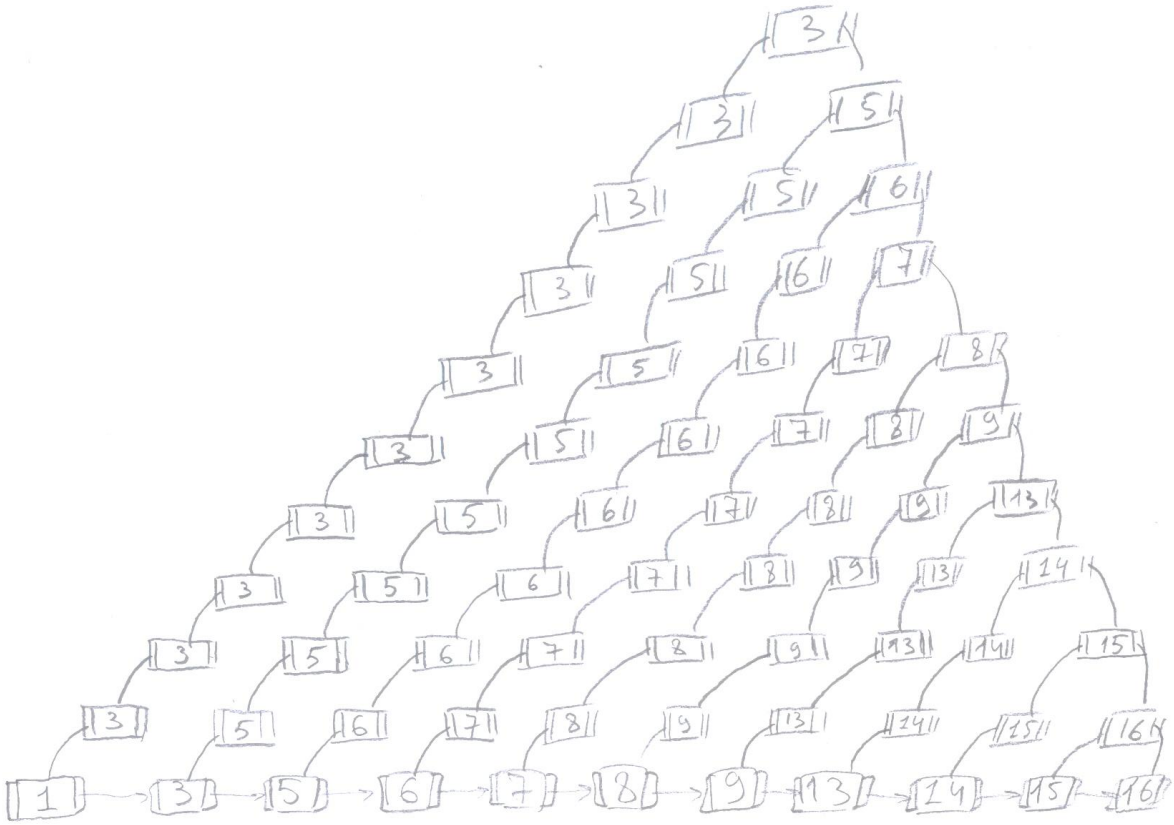
Construct (draw a final result) a B+ tree for n=2 and for the following set of key values {1,3,5,6,7,8,9,13,14,15,16}

Assume that the keys are inserted one by one in the order of their appearance in the list.

According to conditions on number of children and number o values we have:

Non-leaf nodes other than root must have between 1 and 2 children (ceiling(n/2) and n)

Leaf nodes must have between 1 and 1 values (ceiling((n-1)/2) and n - 1)



Q7 [15 pts]

Construct (draw a final result) a B+ tree for n=5 and for the following set of key values

{3, 7, 8, 9, 10, 12, 15, 19, 21, 23, 24, 26, 27, 29, 32, 35, 40, 50, 51, 52}.

Assume that the keys are inserted one by one in the order of their appearance in the list.

According to conditions on number of children and number o values we have:

Non-leaf nodes other than root must have between 3 and 5 children (ceiling(n/2) and n)

Leaf nodes must have between 2 and 4 values (ceiling((n-1)/2) and n - 1)