Bilkent University

Computer Engineering Department

CS351 DATA ORGANIZATION AND MANAGEMENT

Midterm Date: November 6, 2010 Time: 12:10 – 13:55

Student Name/ ID No. Section	**SOLUTIONS**

GOOD LUCK!

Notes: 1. There are 100 points, 6 questions on 7 pages.
2. Please READ the questions. It is a closed book/notes exam.
3. Show your work.
4. You are not allowed to use your cell phone or PDA for any purpose. You cannot share calculators
5. You cannot leave the exam room in the first 30 minutes.
6. Please do not write anything in the following table.

Q. No.	Q1	Q2	Q3	Q4	Q5	Q6	Overall
Points Possible	10	25	22	18	15	10	100
Your Grade							

DISK PARAMETERS: When/if needed use the following disk parameters:

B	block size	2400 bytes
Bkfr	no. of records per block	6
btt	block transfer time	1 ms (data transfer rate is 2400 bytes/ms)
ebt	effective block transfer time	1.2 ms (time needed to read a block in sequential reads)
r	aver. rotational latency time	10 ms
s	average seek time	15 ms

Q1. (10 pts.) An unsorted (pile) file contains 100,000 records of 400 bytes. In every 10 minutes, 3 records are added and 1 record is deleted until the total number of active records is 250,000? You have to show your work give a single number for each question.

a. How long does it take to come to the state of reorganization?
100,00 + 2.x = 250,000; x number of 2 record additions to reach 250,000 records
x= 150,000/2= 75,000 **x=** the no. of 10 min. that involves 3 additions and 1 deletion
➔ total time is then 150,000 min.

b. How long does it take to find a record right before reorganization?
Before reorganization means while having all records including the deleted ones. All deletions are logical. n (no. of records)= 250,000 + 75,000= 325,000
b= (325,000 x 400)/2400= 54,167 (approx.); T_F= ½ . b. ebt= ½ . 54,167 . 1.2= 32,500 ms = 32.5 sec.

c. How long does it take to find a record right after reorganization?
T_F= ½ . (250,000 x 400)/2400 . 1.2= 25,000 ms= 25 sec.

Q2. (25 pts.) Consider a file with size 650 MB, record size 480 bytes. Each bucket contains only one block. Assume that 5 MB of memory is available for sorting and merging. For sorting we want to use heap sort and for merging we want to use 3-way merge. For the file and sorting conditions please answer the following questions.

a. (2 pts) How much time is needed for sorting (i.e., time needed to generate the sorted segments)?

T = 2 * b * ebt $\rightarrow$ we need to calculate b

b = file size / Block size = 650 * 10^6 bytes / 2400 bytes = 270,833.33

(Note that we accepted also the following b value
b = file size / record size * bkfr = 650 * 10^6 bytes / 480 bytes * 6)

$\Rightarrow$ T = 2 * b * ebt = 2 * 270,833.33 * 1,2 ms = 650,000 ms = 650 s

b. (2 pts) How many sorted segments are there after sorting and what cen be said about the size of sorted segments?

# of segments = file size / MM size = 650 MB / 5 MB = 130

We have 130 sorted segments, each of which is 5 MB.

c. (5 pts) During merging how many passes are required? Draw a table that shows the merge pass number, number of segments and the size of each segment?

# of passes = $\lceil \log_3 130 \rceil = 5$

Pass #	1	2	3	4	5
# of sorted segments	130	44	15	5	2
Size of each segment (MB)	130 * 5 MB	43 * 15 MB 1 * 5 MB	14 * 45 MB 1 * 20 MB	4 * 135 MB 1 * 110 MB	1 * 405 MB 1 * 245 MB

d. (3 pts) How much time is required for one pass during merging excluding (s+r)?

T = 2 * b * ebt = 2 * 270,833.33 * 1,2 ms = 650,000 ms = 650 s

(the same T value in part a)

e. (3 pts) How many (s+r) operations are needed for one pass of merge?

2 * p * nsg = 2 * 3 * 130 = 780 (s+r) operations needed

f. (4 pts) How many physical and logical I/O is performed during one pass of merging?

The number of physical I/O is the number of blocks that is read and written.

→ # of phys I/O = 2 * b = 2 * 270,833.33 = 541,666.66

The number of logical I/O is the total number of records that is read and written in each block.

→ # of logical I/O = bkfr * 2 * b

= (block size / record size) * 2 * b

= (2400 bytes / 480 bytes) * 2 * 270,833.33

= 5 * 2 * 270,833.33

= 2,708,333.3

g. (6 pts) Now assume that we double the bucket size. Give the time needed for one pass of merging without considering (s+r) and also state the number of physical I/O and logical I/O.

Doubling the bucket size does <u>not</u> change the time needed for one pass of merging, therefore, it remains the same.

The number of physical I/O is reduced by half of the number of old physical I/O

→ # of phys I/O = old phys I/O / 2 = 541,666.66 / 2 = 270,833.33

The number of logical I/O also remains the <u>same</u>.

Q3. (22 pts.) Consider a linear hashing file environment. The file contains 386 primary area disk blocks. The boundary value is equal to 130. The current load factor of the file is 2/3. The desired load factor of the file is also 2/3. The blocking factor is 12. The following hash function is used to distribute the records among the blocks: Mod (key, 7500)

a. How many bits do we need to use to find the location of a record in this file?

h = ⌊log (number of primary disk blocks)⌋ = ⌊log (386)⌋ = 8

If record's last *h* bits are less than *bv*, then we look at last (h-1) = 9 bits.
Otherwise, look at h = 8 bits.

b. How many disk blocks are at the hashing level h?

Blocks at level h are placed between *bv* and 2^h -1: [bv, 2^h -1]
Then; [130, 2^8 - 1] = [130, 255] = 255-130+1 = 126 blocks

c. How many disk blocks are at the hashing level h+1?

(Number of blocks at level h+1)
= (Total number of blocks) – (Number of blocks at level h)
= 386 – 126 = 260 blocks
or
2 * bv = 2 * 130 = 260 blocks

d. How many records are stored in the file?

Lf = (Number of records in file) / (Number of record positions in prime area)
=> (Number of records in file) = Lf * (Number of record positions in prime area)
(Number of records in file) = 2/3 * (386 * 12)
= 3088 records

e. What can be said about the number of overflow blocks?

We can not state anything on the number of overflow blocks.

f. Assume that we want to insert a record with a key value equal to 150880. In which block are we going to try to insert this record? Express the block number in binary form and show your work.

Hash function: Mod(record, 7500)
Mod(150880, 7500) = 880 is 1101110000 in binary.
Look at last h=8 bits and compare with bv: (01110000=112) < 130
which means use (h+1)=9 bits. Insert into block number: 101110000

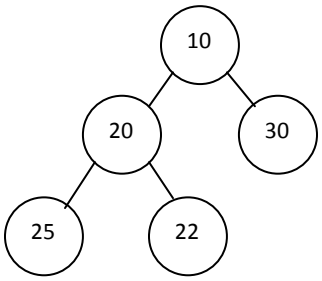
g. Now assume that 20 records are inserted to the file. What are the new values for h and bv? Explain your answer.

We know that we expand for each (Lf * Bkfr) = (2/3 * 12) = 8 records inserted.
20 records = 8 + 8 + 4
which means 2 times we expand; then *bv* becomes 130 + 2 = 132
h does not change.

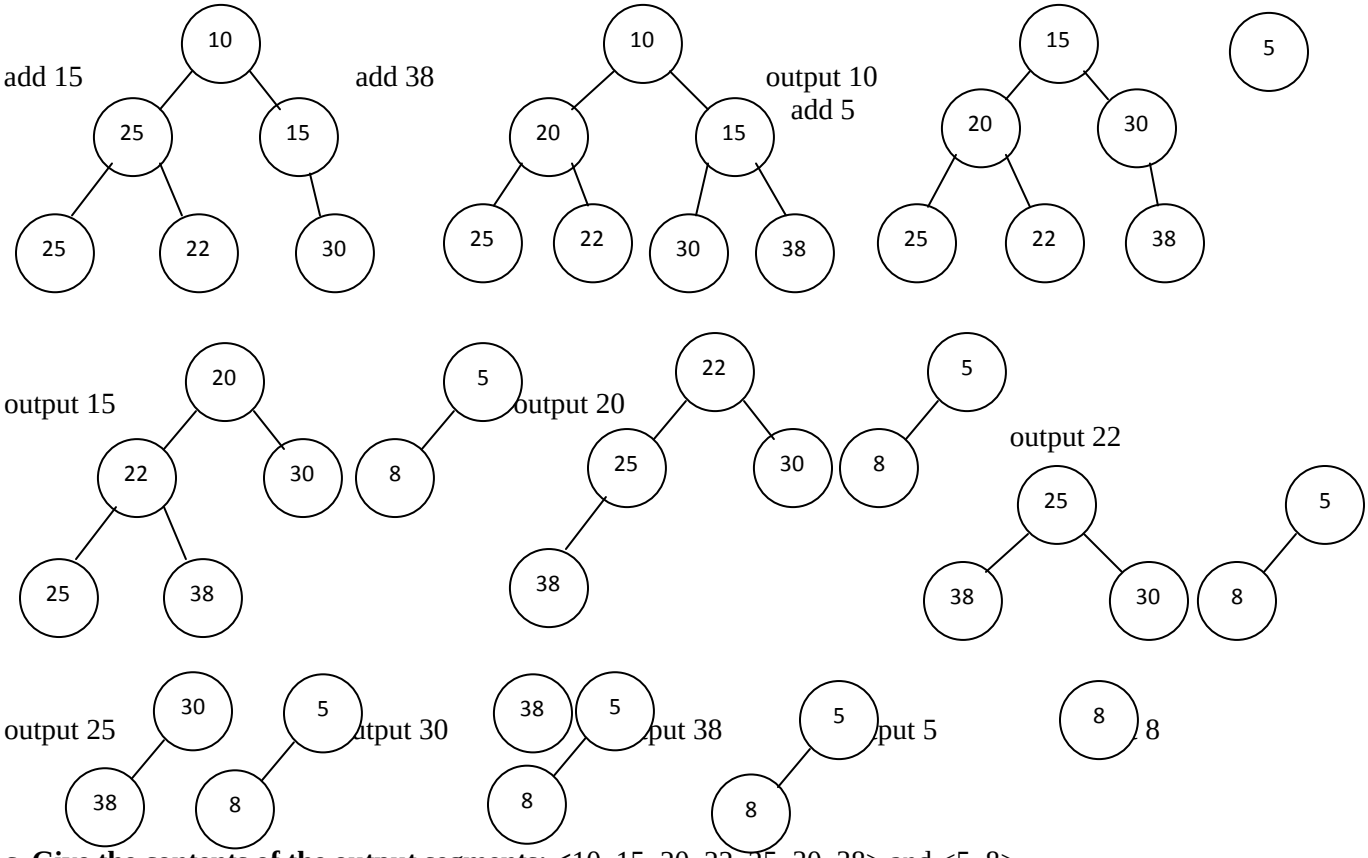
Q4. **(18 pts.)** Consider a replacement selection sort environment. Assume that we have a seven record memory capacity (i.e., total number of records that we can keep in the memory is seven records). So far the memory contains only one heap and its array representation is as follows.

10	20	30	25	22

a. Draw the corresponding tree representation.



b. We want to use the above heap structure continue with replacement selection sort using the records with the following key values 15, 38, 5, and 8. Show the contents of the heap tree structure(s) as we continue with the sorting process.

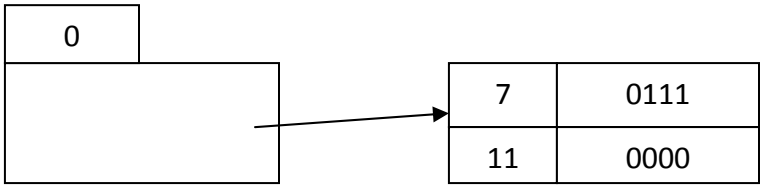


c. Give the contents of the output segments: <10, 15, 20, 22, 25, 30, 38> and <5, 8>

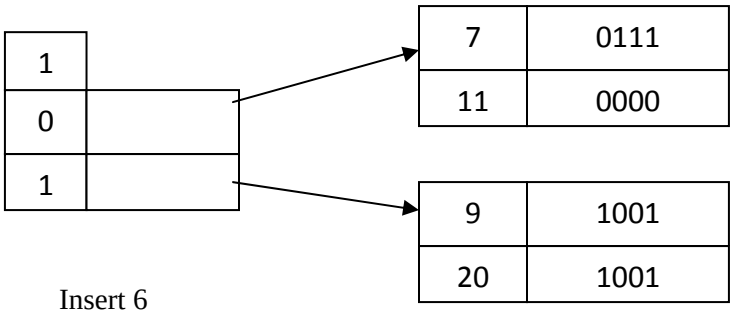
Q5. (15 %) Insert the records with the keys 7 11 9 20 6 12 5 15 into a hash file in the order given using extendible hashing. Use h(k)= key mod 11. Block size is 2 records. Show your steps.

Key	Mod(Key,11)	Bin
7	7	0111
11	0	0000
9	9	1001
20	9	1001
6	6	0110
12	1	0001
5	5	0101
15	4	0100

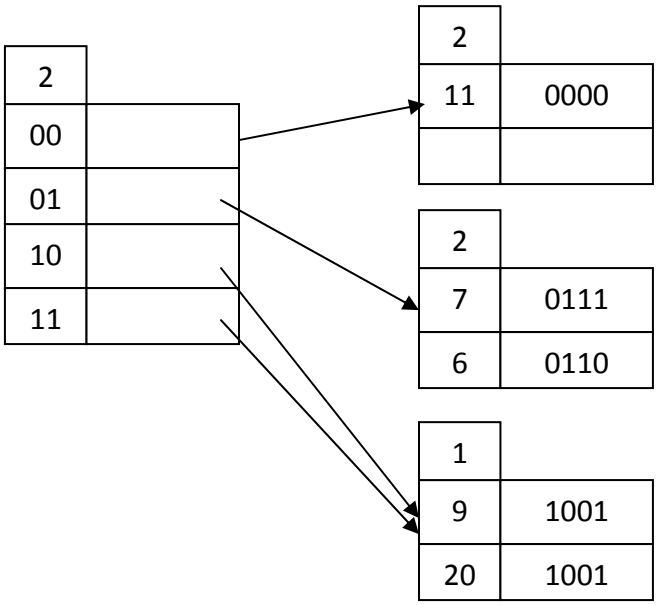
Use prefix!
Insert 7, 11



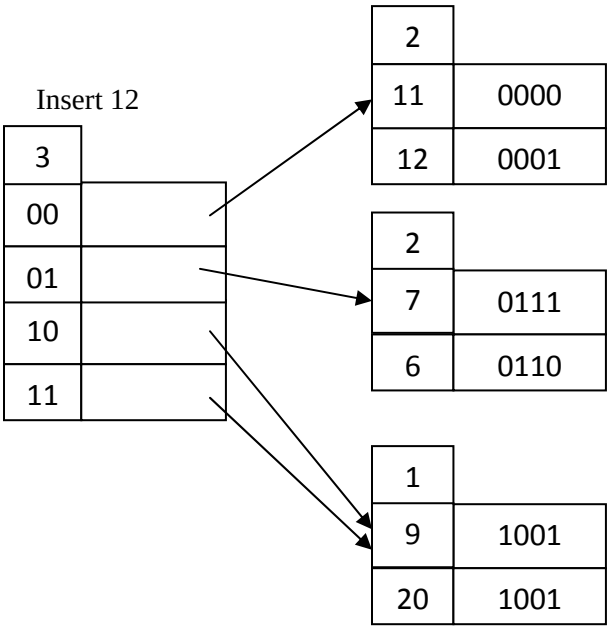
Insert 9, 20



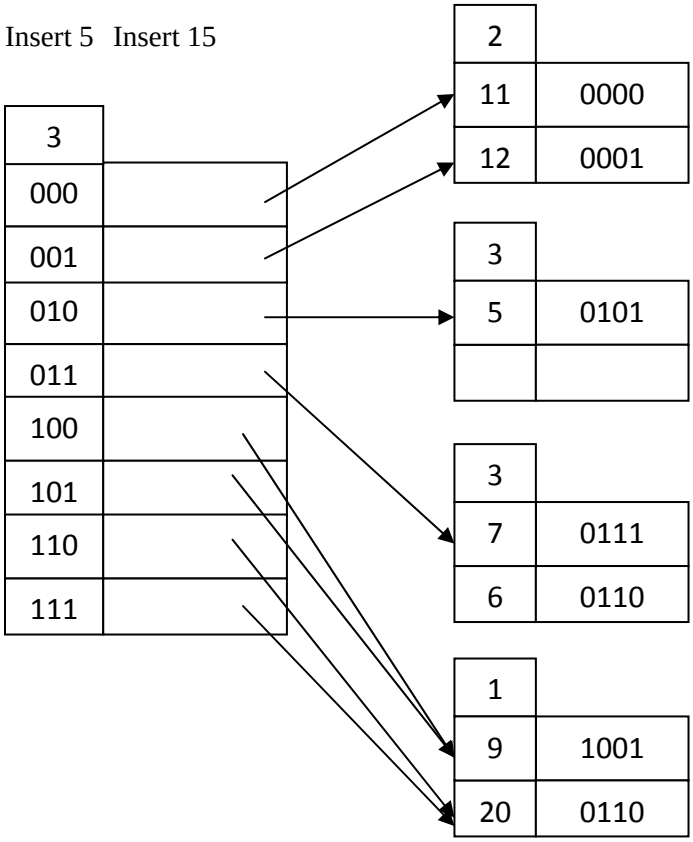
Insert 6

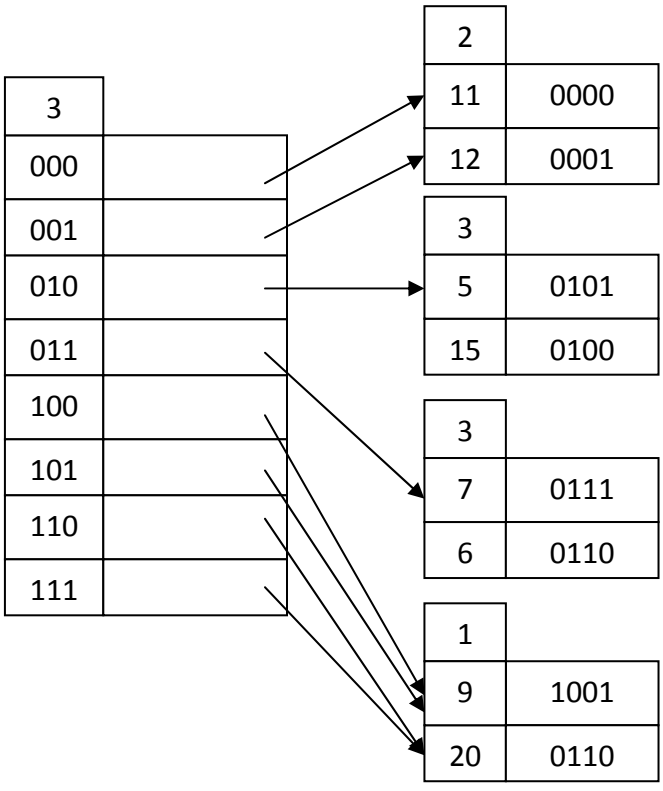


Insert 12



Insert 5 Insert 15





Q6. (10 %) Consider a hard disk unit with the revolution speed of 7200 rpm. The disk contains 5 plates where both surfaces of each plate can contain information. Each surface contains 500 tracks and each track contains 100 blocks. The block size is 2400 bytes.

a. Calculate the rotational latency?

60,000 / 7200= 8.33 ms ➔ r= 4.17 ms (approximate values)

b. What is the number of cylinders?

Assuming that we have one read/write head per surface it is equal to no. of tracks per surface and hence it is equal to 500.

c. How many blocks do we have in a cylinder?

No. of tracks per cylinder= total no. of magnetizable surfaces, hence there are 10 tracks per cylinder and remember that each track contains 100 blocks ➔ 1,000 blocks/cylinder.

d. What is the approximate block transfer time?

One full revolution requires about 8.33 ms and there are 100 blcoks per track and hence it is approximately equal to 8.33 / 100= 0.0833 ms.