First Letter of your last name:___H_________________

Your Full Name: ________Yuichi Hamamoto_______________________

I promise I do not cheat on the exam. Your initial:__Y.H.___

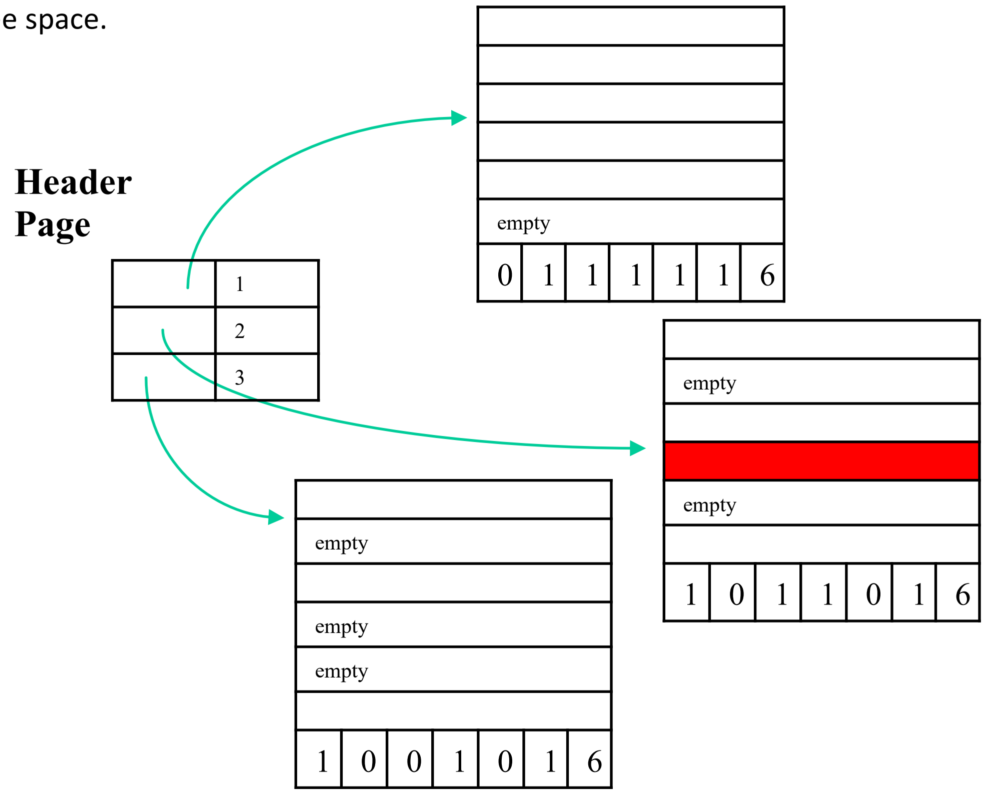
COM S 363: Exam 2

NOTES:

- This is an open book open note exam.
- You must work by yourself, without discussing with anyone else except TAs or instructors.
- You can handwrite, but you are strongly encouraged to type/draw on these slides directly. Some shapes may be given for your convenience. If you do handwrite, write legibly; if the grader can't read, at their sole discretion, you receive no point.
- Answer concisely and precisely. One effective way to lose points is to write nonsenses. HINT: The spaces you are given are more than you need.
- Submit your file in PDF format.

Problem	Max Points	Points
1	15	
2	25	
3	20	
4	15	
5	25	
Total	100	

1. (15 points) Suppose a relation is stored in heap files showing below. The slots with "empty" are free space.



(1) (4 points) From the illustration, can you decide if the records are fix length or variable length, packed or unpacked?

Yes, I can (fixed length&unpacked).

(2) (6 points) Suppose a user inserts another record. What the result files should look like?

The record will be stored into the empty space in the pages because there are still some empty spaces. As a result, the numbers in the header page which indicates the number of available space for the corresponding page will be updated and at the bottom of the page, the binary number for the index will be updated to 1 because no the slot is occupied

(3) (5 points) Suppose a user wants to delete a record. This record is located at the highlighted red slot. What is the procedure?

First it search the record(find the page from the header page then go through the page). Then it will delete the record(overwrite).

2. (25 points) Consider a relation with this schema, Employee(ename:string, eid:integer, age:integer, salary:integer). Suppose ename is indexed by a sparse B^+-tree (primary) and eid is indexed by a dense and unclustered B^+-tree (secondary), as roughly illustrated by the figure showed in the next page. Suppose the relation has 1,000 pages, and each page is 512 bytes and can hold 16 records. Size of primary search key is 23 bytes, size of secondary search key is 2 bytes, and the size of a pointer is 2 byte.

a) (5 points) Calculate the number of pages for data entry nodes in the secondary B^+-tree.

2n+(n+1)2<=512
N<=127.5
Therefore, with secondary key one node can point to 128 nodes
Dense index, one pointer per one record
16,000 pointers on the leaf level
16,000/128=125 pages on the leaf level(data entry nodes)

Estimate the total number of pages that need to be read from the disk in order to answer the following queries. Explain your answer. For each query, you can assume the selection factor is 0.1, i.e., out of 1,000 * 4 records, 1,000 * 4 * 0.1 = 400 records will satisfy each query condition.

b) (5 points) Find all employees whose ages are in between 40 and 50.

Because the age is not indexed, we need to read the entire relation.
Therefore, the total number of pages that need to be read from the disk in order to find all employees whose ages are in between 40 and 50 is 1000 pages.

c) (7 points) Find all employees whose name starts with a character that is in between “C” and “F”.
Sparse index, one pointer per page
23n+(n+1)2<=512
N<=20.4
Therefore, with primary key one node can point to 21 nodes
1,000/21=48 pages on the leaf level(data entry nodes)
The tree is 3 level
48+3+1 = 52 page, the B+tree will take.
Read tree: 2 or 3 index pages
Therefore, the total number of pages that need to be read from the disk in order to find all employees whose name starts with a character that is in between “C” and “F” is 3 pages with best case and 52 page with the worst case.

d) (8 points) Find all employees whose eid is in between 100 to 200.

Dense index, one pointer per one record

1000*16*0.1 =1,600 records

2n+(n+1)2<=512

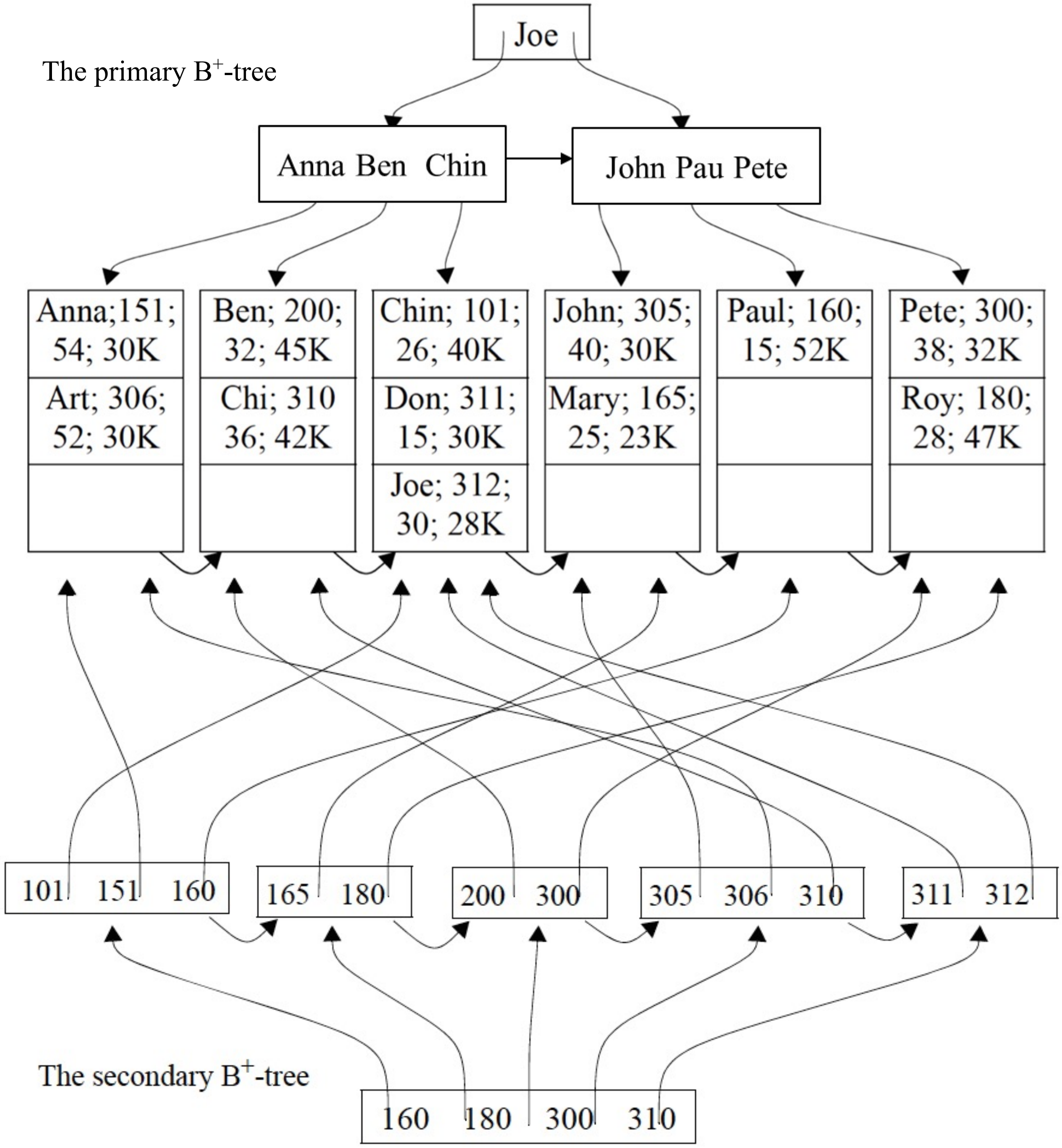
N<=127.5

Therefore, with primary key one node can point to 128 nodes

1,600/128=13 pages on the leaf level(data entry nodes)

The tree is 1 level

Therefore, the total number of pages that need to be read from the disk in order to find all employees whose eid is in between 100 to 200 is 1 page with the bast case and 13 pages with the worst cases.


The primary B+-tree
Joe
Anna Ben Chin
John Pau Pete
Anna;151; 54; 30K
Art; 306; 52; 30K
Ben; 200; 32; 45K
Chi; 310 36; 42K
Chin; 101; 26; 40K
Don; 311; 15; 30K
Joe; 312; 30; 28K
John; 305; 40; 30K
Mary; 165; 25; 23K
Paul; 160; 15; 52K
Pete; 300; 38; 32K
Roy; 180; 28; 47K
101 151 160
165 180
200 300
305 306 310
311 312
The secondary B+-tree
160 180 300 310

3. (20 points) Explain how to sort 120 pages of data (e.g., numbers) with 4 pages of main memory in detail (i.e., pass 1, pass 2, ..., until the data is sorted) and estimate the I/O cost. Your algorithm needs to minimize disk I/Os (i.e., the number of pages that need to be read from and written to the disk). When sorting in main memory, you can assume you have the additional memory for swapping two numbers.

Using merge sort,
Read 3 page from the 120 pages and sort and merge into a sub list
Then we will have 40 sub list and do the same operation.
Then we will have 14 sub list and do the same operation.
Then we will have 5 sub list and do the same operation.
Then we will have 2 sub list and do the same operation.
Then now the pages are sorted.
Therefore, I/O cost is <u>2*120*6=240*6=1440</u>

4. (15 points) Suppose we have relation R of 120 pages in size, and relation S of size 40 pages, not sorted. We want to join R and S based on their primary keys. The size of main memory is 10 pages. Suppose the hash functions we choose uniformly partition the relation, fudge factor f=1.

   a) (6 points) After the partition phase, (numeric answers)

      R is divided into 8(10-2) partitions, each of which is about 15 pages;

      S is divided into 8(10-2) partitions, each of which is about 5 pages.

   b) (3 points) In the probing phase, which relation should be further partitioned using the B-2 memory pages?
      i. R
      ii. S
      iii. Both R and S
      iv. Either R or S
      v. Neither R nor S

      your answer: i

   c) (3 points) Suppose the memory size is 5 pages instead of 10 pages. Is this change going to affect the I/O cost of Grace Hash Join? Choose your answer and explain why.
      i. Yes, I/O cost is going to increase
      ii. Yes, I/O cost is going to decrease
      iii. No, I/O cost is going to remain the same
      iv. It may or may not affect the I/O cost.

      your answer:i

      explanation: Because the relations need to be partitioned further due the decrement in the memory size, the partition cost increase. Therefore, the total cost will increase.
      5<sqrt(40) which now does not satisfy B>sqrt(fM), so we need to partition S further.

d)(3 points) Suppose the memory size is 15 pages instead of 10 pages. Is this change going to affect the I/O cost of Grace Hash Join? Choose your answer and explain why.

i. Yes, I/O cost is going to increase
ii. Yes, I/O cost is going to decrease
iii. No, I/O cost is going to remain the same
iv. It may or may not affect the I/O cost.

your answer:ii

explanation: Similar to c, because the relation R does not need to be partitioned further due the increment in the memory size, the partition cost increase. Therefore, the total cost will increase. 15>sqrt(120) which now satisfies B>sqrt(fM), so we do not need to partition R further.

5. (25 points) Derive the I/O costs of different join algorithms of relations R and S given the following variables, which you may or may not use all of them. Ignore the CPU time costs and the cost of writing the results. Write down steps for partial credits

    |R|=10: Number of tuples in R
    |S|=20: Number of tuples in S
    M=120: Number of pages in R
    N=40: Number of pages in S
    B=10: Number of available memory in pages

a) (5 points) What is the minimal I/O cost of block nested loop join?

If R is outer: 120+ceiling(120/(10-2))*40=120+15*40=720
If S is outer: 40+ceiling(40/(10-2))*120=40+5*120=640
<u>The minimal I/O cost is 640</u>

a) (5 points) What is the minimal I/O cost of simple nested loop join?

If R is outer: 120 +120*10*40 =48120
If S is outer: 40 +40*20*120 = 96040
<u>The minimal I/O cost is 48120</u>

a) (5 points) What is the minimal I/O cost of indexed nested Loops Join? (suppose the cost of retrieving a matching tuple is 2, for both R and S)

For R: 120+120*10*2 = 2520
For S: 40 + 40*20*2 = 1640
<u>The minimal I/O cost is 1640</u>

a) (5 points) What is the minimal I/O cost of grace hash join?

Total cost is 3(M+N) = 3(160)=<u>480</u>

a) (5 points) What is the minimal I/O cost of Sort-Merge Join? (suppose the join is on their primary keys which are sorted already)

Because here we do not need to sort, the total cost is simply the merging cost.
Therefore M + N = <u>160</u>