Homework 1

1.Consider the instance of the Students relation shown below, sorted by age. Assume that these tuples are stored in a sorted file in the following order: the first 3 tuples are on page 1 and the last 3 tuples are on page 2 (each page can store up to 3 data records)

Sid	Name	Login	Age	gpa
53831	Madayan	madayan@music.com	11	1.8
53832	Guldu	guldu@music.com	12	2.0
53660	Mary	Mary@cs.com	13	3.0
53666	Jones	jones@cs.com	18	3.4
53688	Smith	smith@ee.com	19	3.2
53650	Smith	smith@math.com	19	3.8

Explain the data entries in each of the following indexes contain. If such an index cannot be constructed, say so and explain why.

1) An unclustered index on age using Alternative (1)
2) An unclustered index on age using Alternative (2)
3) An unclustered index on age using Alternative (3)
4) An clustered index on age using Alternative (1)
5) An clustered index on age using Alternative (2)
6) An clustered index on age using Alternative (3)
7) An unclustered index on gpa using Alternative (1)
8) An unclustered index on gpa using Alternative (2)
9) An unclustered index on gpa using Alternative (3)
10) An clustered index on gpa using Alternative (1)
11) An clustered index on gpa using Alternative (2)
12) An clustered index on gpa using Alternative (3)

---

2. Consider a disk with the following characteristics (these are not parameters of any particular disk unit): sector size B=512 bytes, each block contains 15 sections and the number of sections per track=450, number of tracks per surface=400. A disk pack consists of 20 double-sided disks.

(a) How many cylinders are there?

(b) What is the capacity of a track, a cylinder and the disk?

(c) Suppose the disk platters rotate at a speed of 2400 revolutions per minute), what is the average rotational delay?
If one track of data can be transferred per revolution, what is the transfer rate?

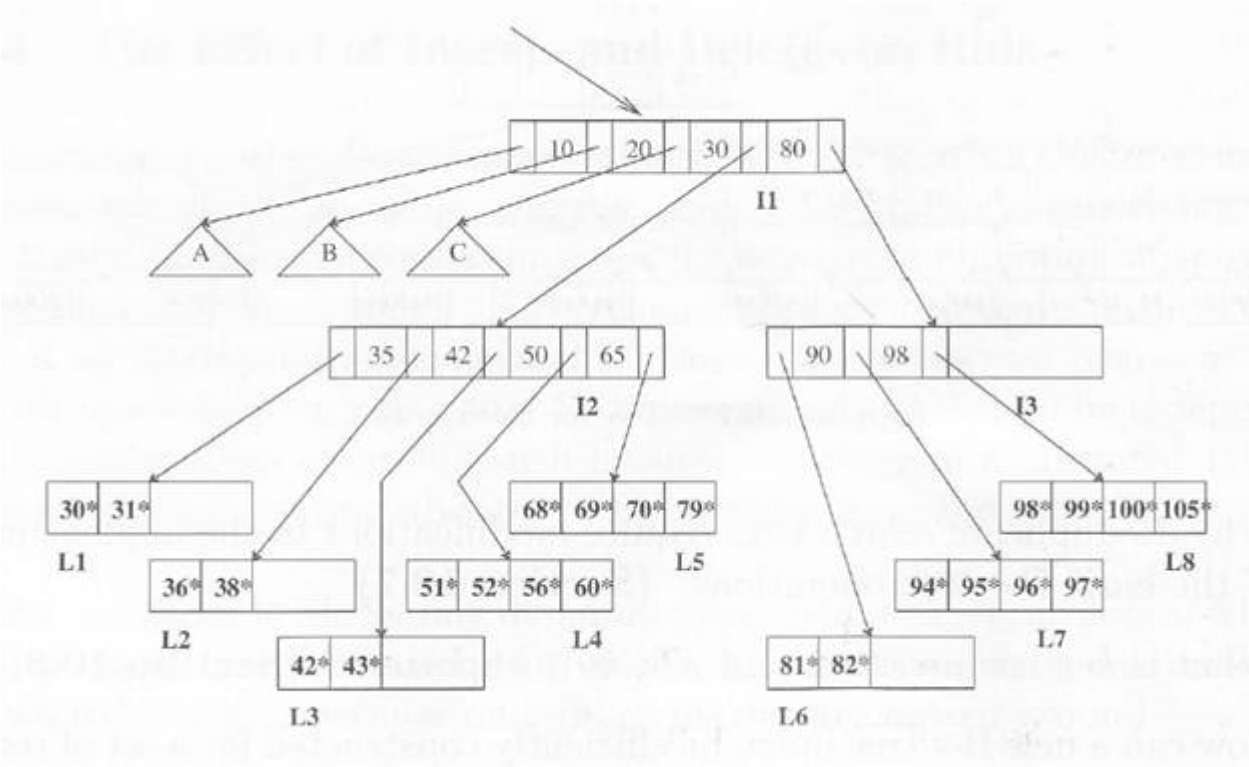
(d) Suppose the average seek time is 30 msec. How much time does it take (on the average) to locate and transfer a single block given its block address?

(e) Calculate the average time it would take to transfer 30 random blocks.

(f) Calculate the time it would take to transfer 30 consecutive blocks.

---

3. **Consider the B+ tree index shown in the following figure, which uses Alternative 1 for data entries. Each intermediate node can hold up to five pointers and four key values. Each leaf can hold up to four records, and leaf nodes are doubly linked as usual, although these links are not shown in the figure. Answer the following questions.**



1) Name all the tree nodes that must be fetched to answer the following query: "Get all records with search key greater than 32 and less than 81"

2) Insert a record with search key 91 into the tree

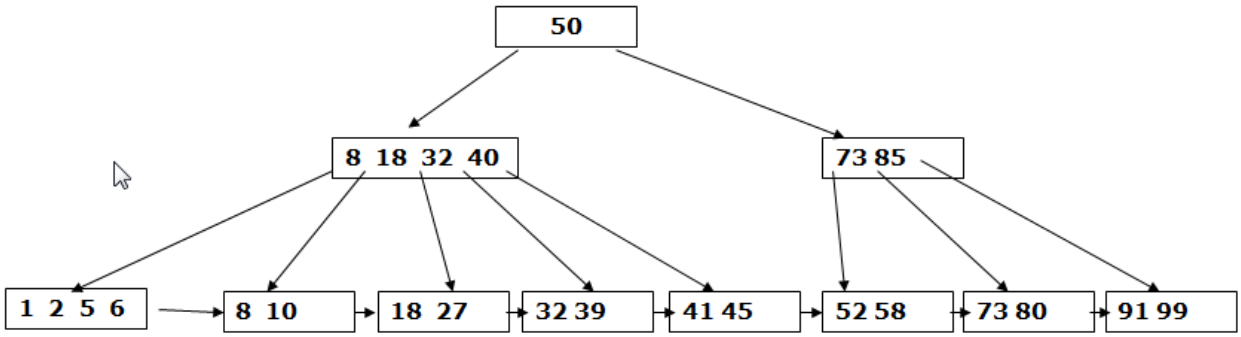
3) Delete the record with search key 30 from the original tree

4) Name a search key value such that inserting it into the original tree would cause an increase in the height of the tree

5) Suppose that this is an ISAM index. What is the minimum number of insertions needed to create a chain of two overflow pages? Give an example of such insertions.

---

4. Consider the B+ tree shown in the following figure. Show the structure of the new tree after the following insertions and deletions.



1) **Insert 9***
2) **Insert 11* after 1)**
3) **Delete 10* from the original tree**
4) **Delete 52* after 3)**
5) **Successively delete 6, 18, 27, 41, 45, 39, 58 after 4)**

---

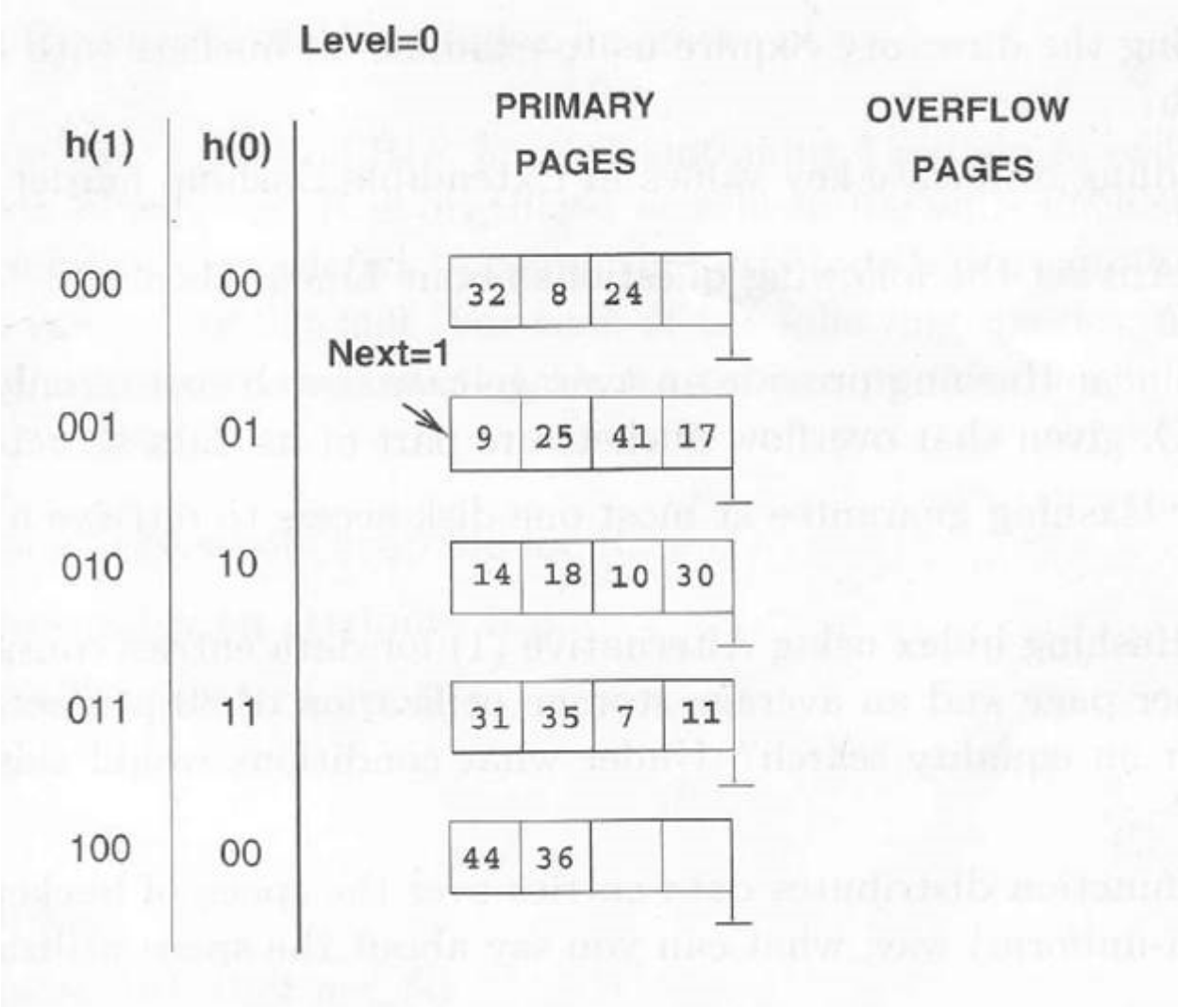
5. Assume that you have just built a dense B+ tree index using Alternative 2 on a heap file containing 40,000 records. The key field for this B+ tree index is 20-bytestring, and it is a candidate key. Pointers (i.e., record ids and page ids) are (at most) 10-byte values. The size of one disk page is 2000 bytes. The index was built in a bottom-up fashion using the bulk loading algorithm, and the nodes at each level were filled up as much as possible.
1) How many levels does the resulting tree have?
2) For each level of the tree, how many nodes are at that level?
3) How many levels would the resulting tree have if key compression is used and it reduces the average size of each key in an entry to 10 bytes?
4) How many levels would be the resulting tree have without key compression but with all pages 70% full?

---

6. Consider the Linear Hast hashing index shown in the following figure. Assume that we split whenever an overflow page is created. Answer the following questions about this index:

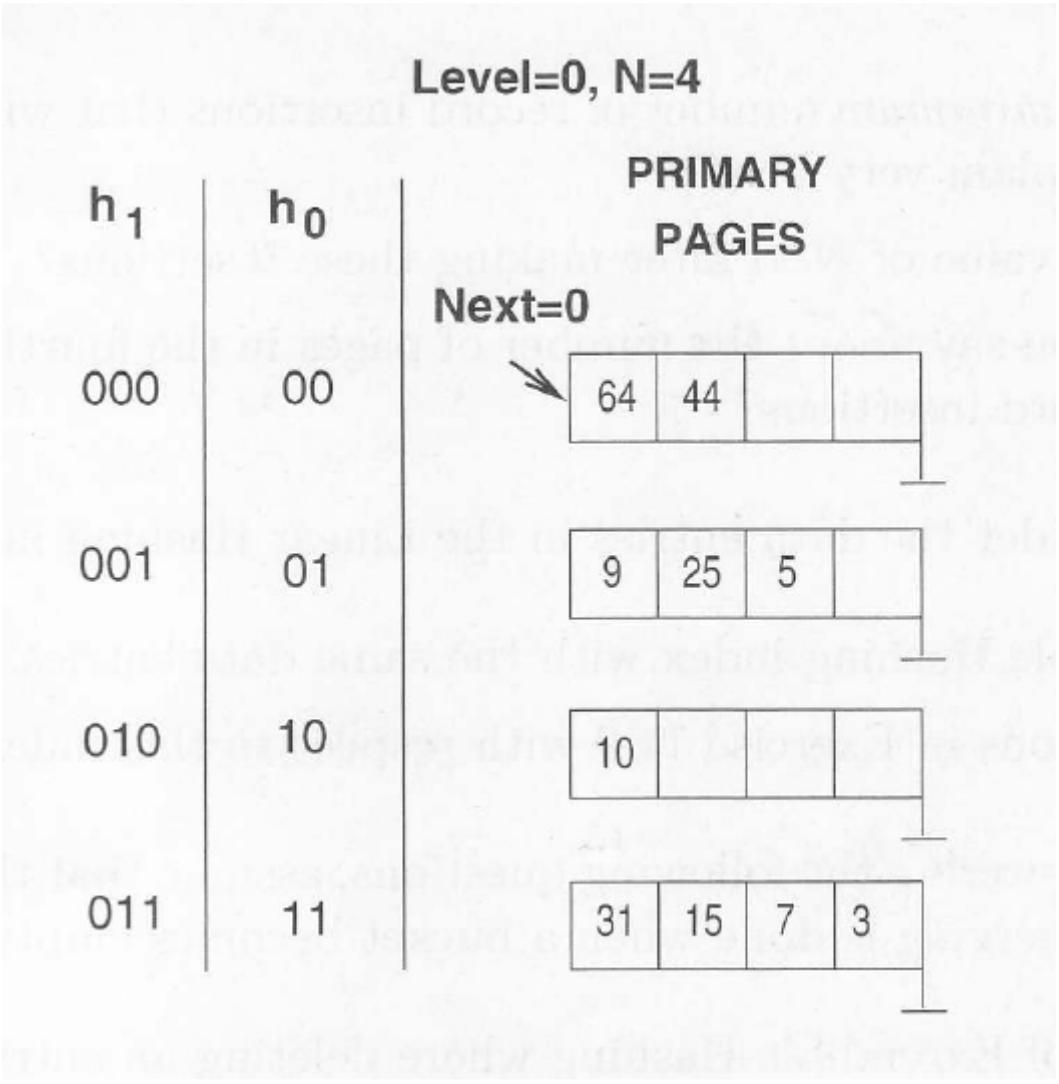
1) Show the index after inserting an entry with hash value 12.
2) Show the original index after inserting an entry with hash value *29*.
3) Show the original index after deleting the entries with hash values 36 and 44. (Assume that the full deletion algorithm is used).
4) Find a list of entries whose insertion into the original index would lead to a bucket with two overflow pages. Use as few entries as possible to accomplish this.

What is the maximum number of entries that can be inserted into this bucket before a split occurs that reduces the length of this overflow chain?



---

7. Consider the data entries in the Linear Hashing index shown in the following figure. Assume that a bucket split occurs whenever an overflow page is created.
1) What is the maximum number of data entries that can be inserted (given the best possible distribution of keys) before you have to split a bucket? Explain very briefly.
2) show the index after inserting a single record whose insertion causes a bucket split.
3) a. What is the minimum number of record insertions that will cause a split of all four buckets? Explain very briefly.
b. What is the value of Next after making these insertions?
c. What can you say about the number of pages in the fourth bucket after this series of record insertions?


Level=0, N=4
h 1
h 0
PRIMARY
PAGES
Next=0
000
00
64
44
001
01
9
25
5
010
10
10
011
11
31
15
7
3