Name: Rohan Maharjan

09/04/2018

17.18

g.

Suppose that the file is not ordered by the key field Ssn and we want to construct a B+-tree access structure (index) on Ssn. Calculate (i) the orders p and p leaf of the B+-tree; (ii) the number of leaf-level blocks needed if blocks are approximately 69% full (rounded up for convenience); (iii) the number of levels needed if internal nodes are also 69% full (rounded up for convenience); (iv) the total number of blocks required by the B+-tree; and (v) the number of block accesses needed to search for and retrieve arecord from the file—given its Ssn value—using the B+-tree.

i.

➔ For order of non-leaf nodes:
- p(P) + (p-1)(Vssn) <= 512
- p (6) + (p-1)(9) <= 512
- 6p + 9p – 9 <= 512
- 15p <= 521
- P <= 34.73
- Therefore, order (p) = 34

➔ For order of leaf node:
- p(Vssn + P_R) + P <= 512
- p(9 + 7) + 6 <= 512
- 16p + 6 <= 512
- 16p <= 506
- p <= 31.62
- Therefore, order (p_{leaf}) = 31

ii.

➔ 0.69 * p_{leaf} = 0.69 * 31 = 22. There will be 22 search keys per leaf node. The file is not ordered on that field so there will have to be a search record for each record in the file, i.e 30,000.

➔ Number of leaf blocks = ceiling(30,000/ 22)
= 1364 blocks

iii.

➔ 0.69 * p = 0.69 * 34 = 24. There will be 24 block pointers per internal node

➔ There are 1364 blocks in the first level of the tree, so there must be 1364 block pointers in the second level.
- 1364/24 = 57 blocks in the second level.

➔ Similarly, in the third level,

- 57/24 = 3 blocks

➔ Finally, in the fourth level

- 3/24 = 1 block

➔ Therefore, the number of levels needed is 4.

iv. 1st level = 1364
2nd level = 57
3rd level = 3
4th level = 1
Therefore, the total blocks = 1364+57+3+1 = 1425 blocks.

v. 4 accesses are needed to traverse the levels. We also need 1 more access to get the primary record from the file. Therefore, 5 total block accesses

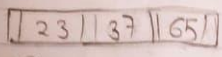
17.19

A PARTS file with Part# as the key field includes records with the following Part# values: 23, 65, 37, 60, 46, 92, 48, 71, 56, 59, 18, 21, 10, 74, 78, 15, 16, 20, 24, 28, 39, 43, 47, 50, 69, 75, 8, 49, 33, 38. Suppose that the search field values are inserted in the given order in a B+-tree of order p = 4 and pleaf = 3; show how the tree will expand and what the final tree will look like.

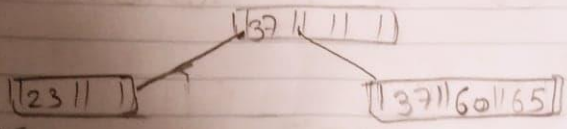
→Please look below:

17.19 P = 4, Pleaf = 3

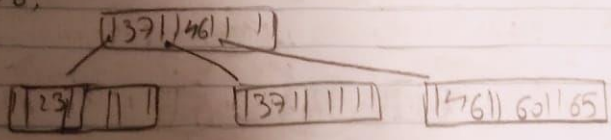
- Insert 23, 65, 37



- Insert 60,

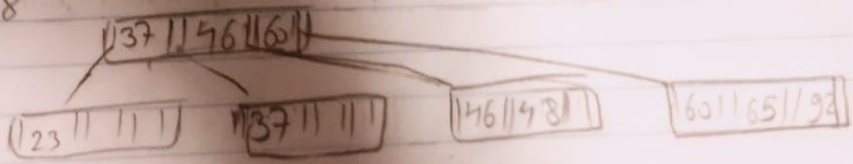


- Insert 46,

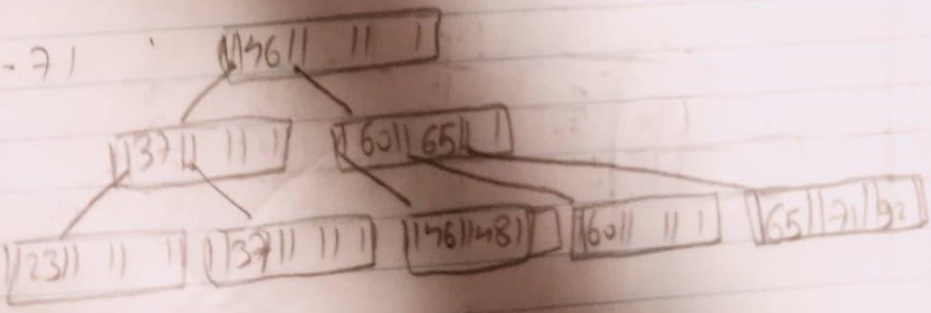


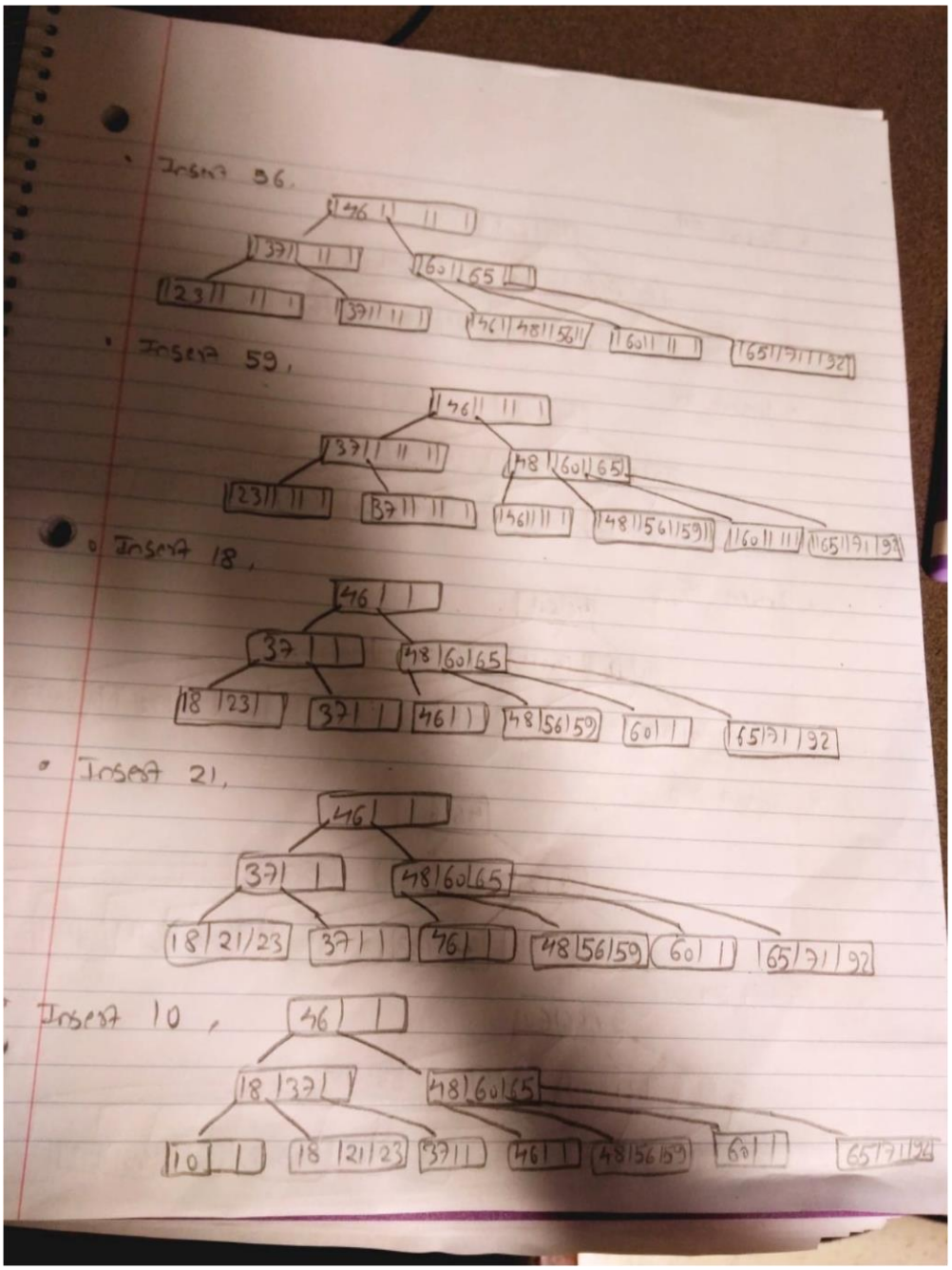
- Insert -92,

- Insert -48

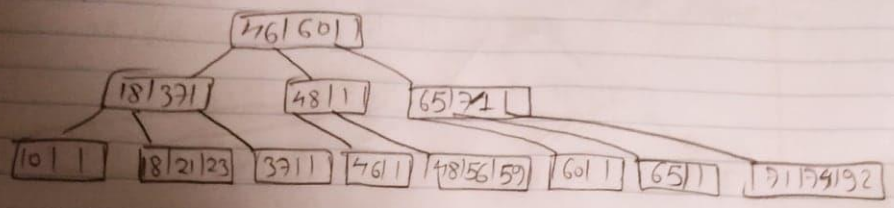


- Insert -71

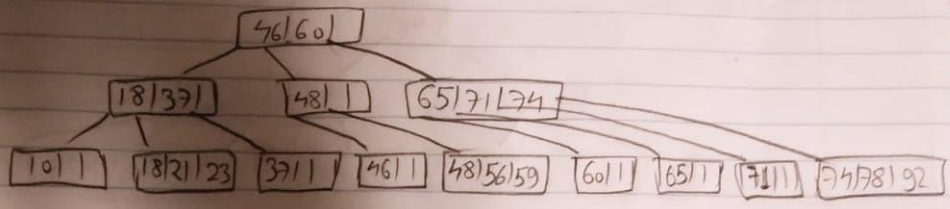



Insert 56.
46
37
60 65
23
39
46 48 56
60
65 71 32
Insert 59.
46
37
48 60 65
23
37
46
48 56 59
60
65 71 32
Insert 18.
46
37
48 60 65
18 23
37
46
48 56 59
60
65 71 32
Insert 21.
46
37
48 60 65
18 21 23
37
46
48 56 59
60
65 71 32
Insert 10.
46
18 37
48 60 65
10
18 21 23
37
46
48 56 59
60
65 71 32

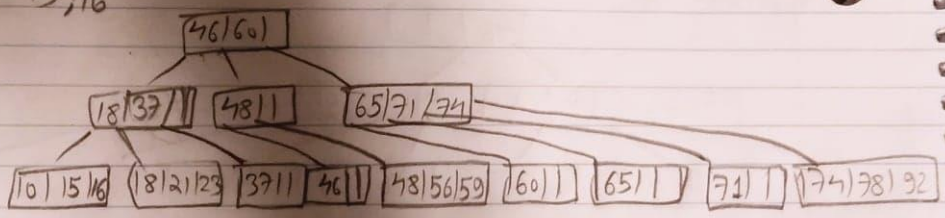
- Insert 74



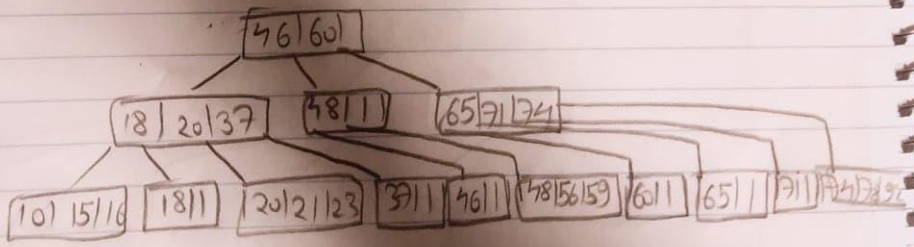
- Insert 78



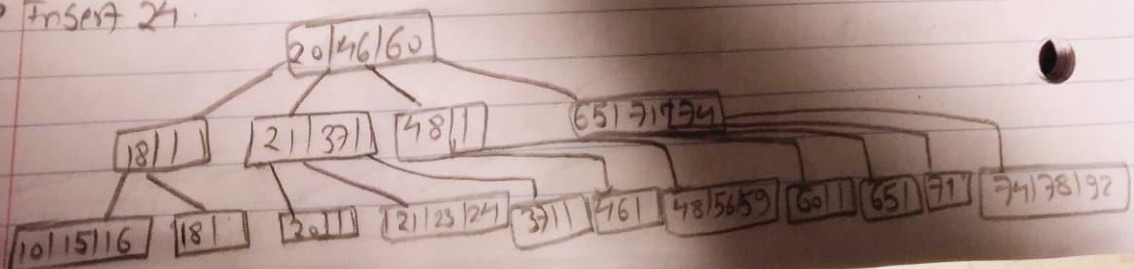
- Insert 15, 16

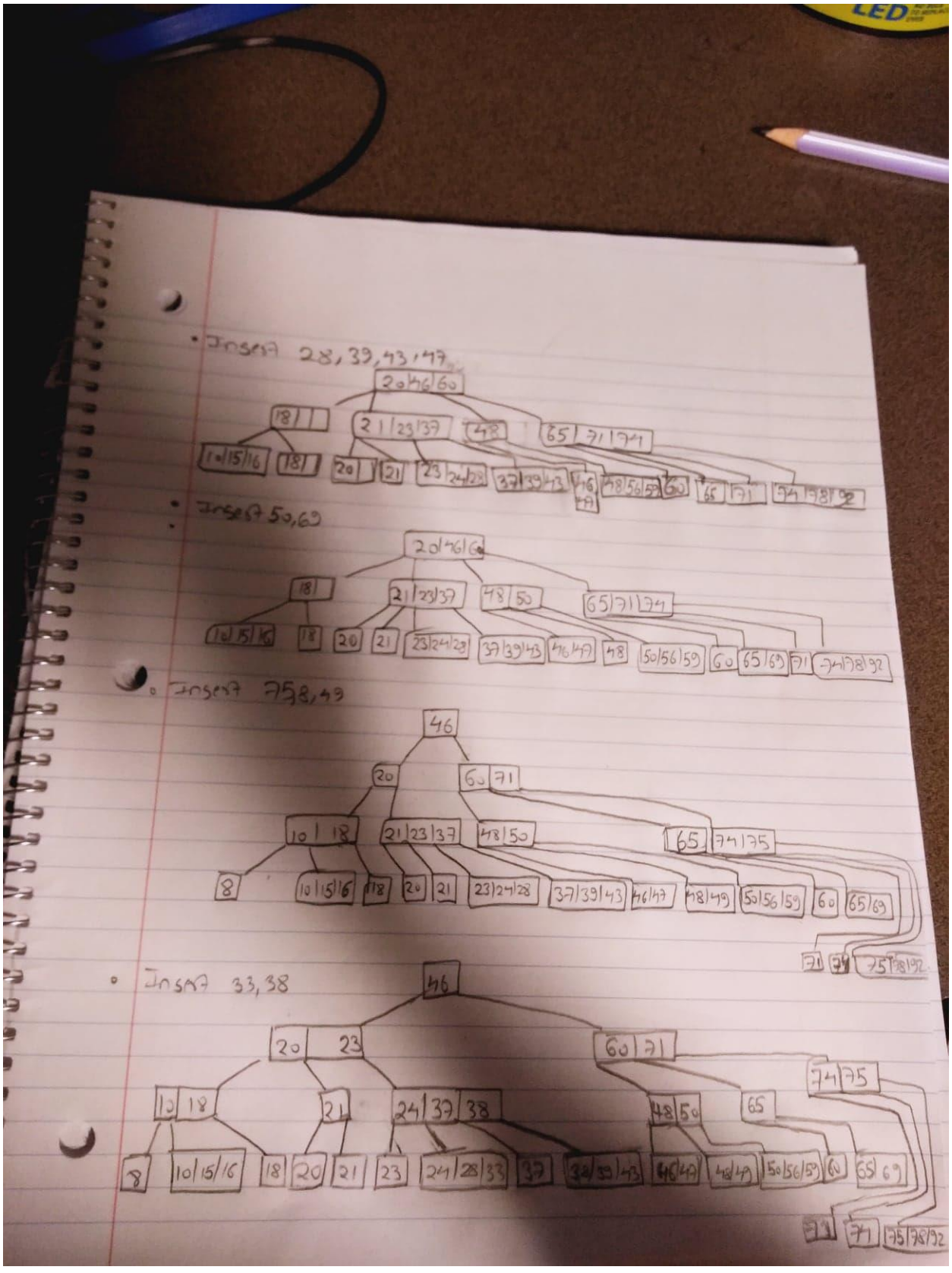


- Insert 20



- Insert 24




• Insert 28, 39, 43, 47
• Insert 50, 60
• Insert 75, 8, 49
• Insert 33, 38