Q1)

a.) after deletion k = 66 and insertion k = 5, 9 and 12

Tree diagram:
- Root: 45
- Internal node (left): 9 | 20
- Internal node (right): 55 | 81
- Leaves: 5 | 7 → 9 | 11 | 12 → 20 | 32 → 45 | 51 | 52 → 55 | 72 → 81 | 82 | 90
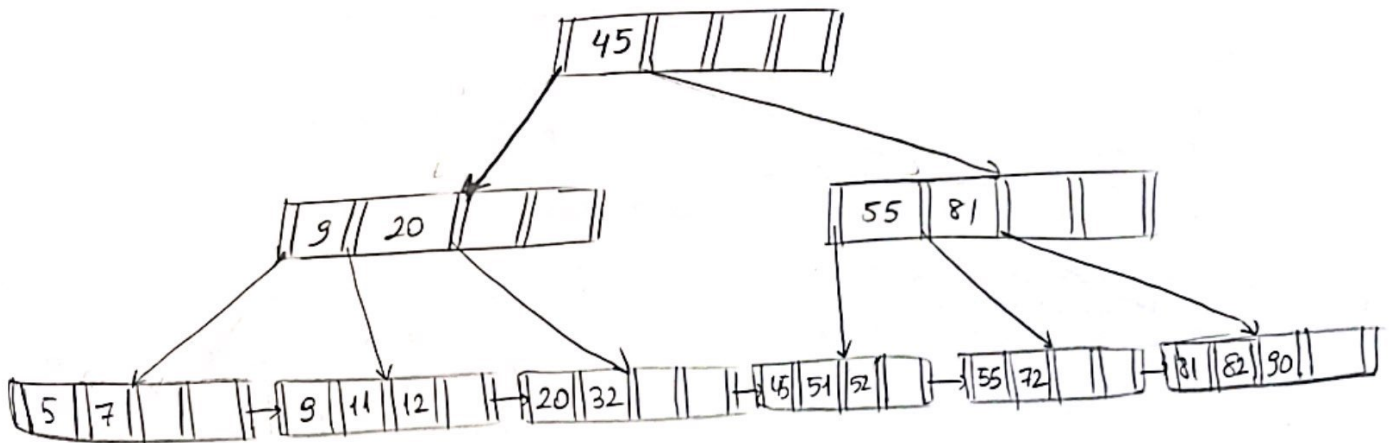
b) Insertion k = 10, 15 and deletion k = 20 (from original tree)

Tree diagram:
- Root: 15 | 45 | 66 | 81
- Leaves: 7 | 10 | 11 → 15 | 32 → 45 | 51 | 52 | 55 → 66 | 72 → 81 | 82 | 90
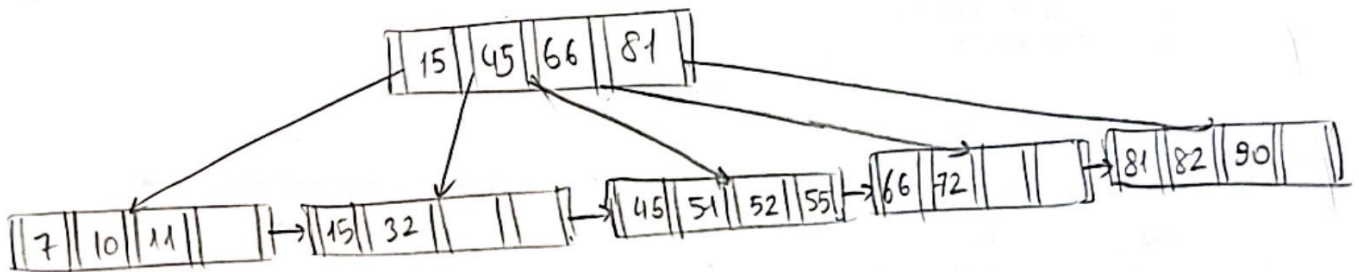
c) insertion k = 85, 95 (into original tree)

Tree diagram:
- Root: 66
- Internal node (left): 20 | 45
- Internal node (right): 81 | 85
- Leaves: 7 | 11 → 20 | 32 → 45 | 51 | 52 | 55 → 66 | 72 → 81 | 82 → 85 | 90 | 95
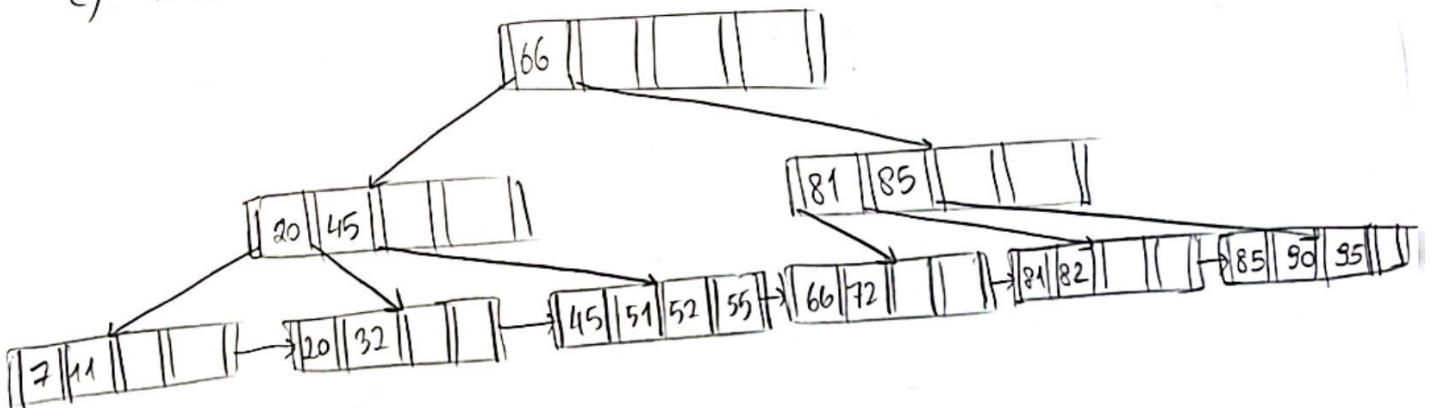
d) deletion k=11 (from the resulting tree in part-c)
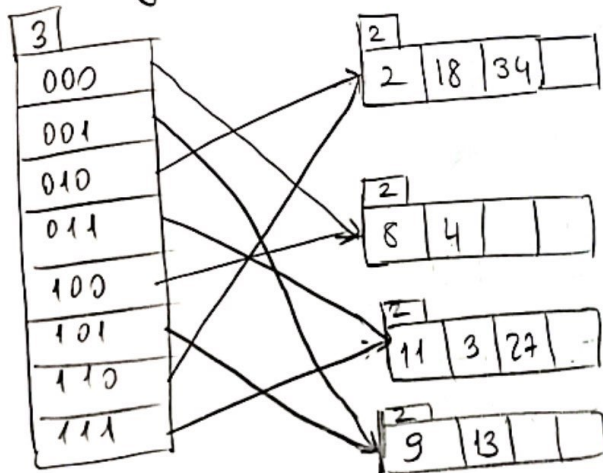
İpek
Öztaş
22003250
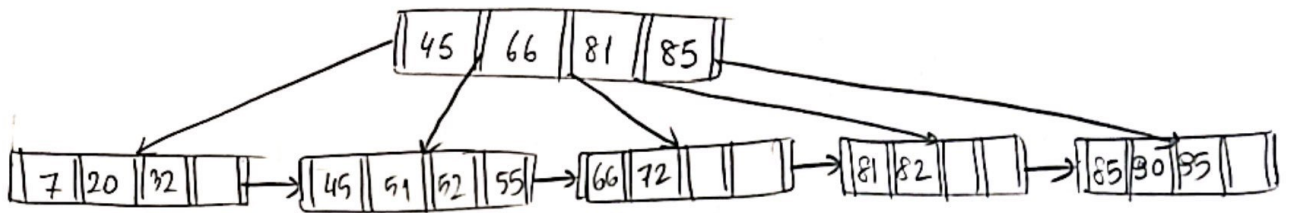
Q2)

a) min. number of key value insertions → $\boxed{2}$

To cause the bucket address table to double, we need to insert a key value that results in a new bit being added to the bucket address table.

This means that the least significant bit of the hash value of the inserted key should be different from the existing bit values.

One such sequence of insertions:

1- Insert key value 4.   (1<u>00</u>)
2- Insert key value 13.   (1 1<u>01</u>)

3
000
001
010
011
100
101
110
111

| 2 | 18 | 34 | |

| 2 |
| 8 | 4 | | |

| 2 |
| 11 | 3 | 27 | |

| 2 |
| 9 | 13 | | |

b) Max. number of insertions = $\boxed{24}$

When the bucket address table doubles, a new bit is added, so there are $2^3 = 8$ addresses. Each can direct to one bucket which can contain 4 key values at most. $8 \times 4 = 32$ key values are possible. There are already 8 key values in B1 and B2. Thus, max. number of insertions is $32 - 8 = \boxed{24}$.

One such sequence of insertions:

1- Insert 4.
2- Insert 13.
3- Insert 17
4- Insert 21
5- Insert 12
6- Insert 16
7- Insert 38
8- Insert 31
9- Insert 42
10- Insert 30
11- Insert 22
12- Insert 14

13- Insert 20
14- Insert 28
15- Insert 36
16- Insert 44
17- Insert 39
18- Insert 47
19- Insert 55
20- Insert 19
21- Insert 25
22- Insert 29
23- Insert 33
24- Insert 37

Final contents: