A 1) Onur Yaman

2) 2007961

B

3) True

4) True

5) True

6) False

7) True

8) True

9) False

10) True

C

11) proper

12) O(h)

13) O(n·logn)

14) True

15) O(n)

16) cycle

D

17) Complete binary tree (of heap data structers)
is a binary tree such that

- At all level d where $0 \leq d \leq h-1$ each node has two children
- At last level, each nodes are in leftmost position.

18) "Stable" means, after sorting, positions of all pairs whose keys are equal must be same position.
For example, (3,d), (2,c), (2,b), (1,a)
after sorting, (1,a), (2,c), (2,b), (3,d)
it is stable

E

19) A and C

20) min, remove_min, add, first, last

21) radix sort

F

22)
```
preorder(T, p):
    visit(p)
    for each c in T.children(p)
        if not c visited
            visit(c)
            preorder(T, c)
```

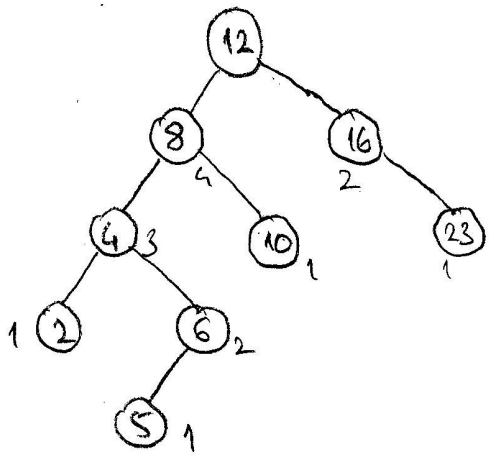
G

23) (a) 5, 6, 23, 16, 20, 18

(b) left child of node 18

24) (a) Balanced

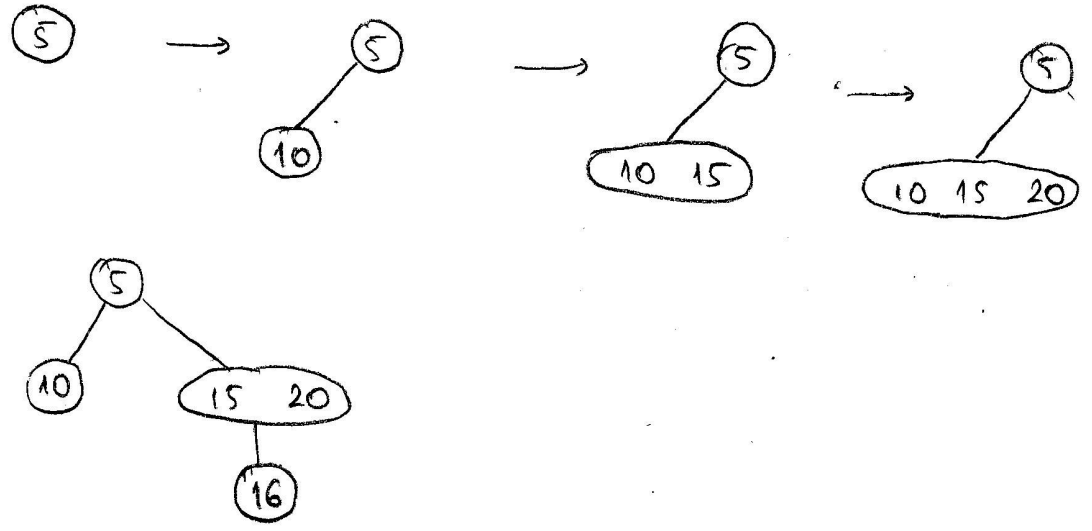
(b) Insert 5, then it will be <u>unbalanced</u>.



The pivot is 8.

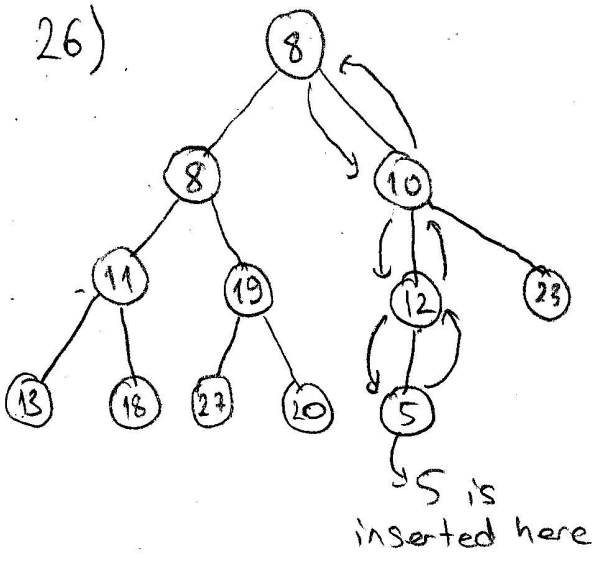
Case 2, double rotation should be applied. Firstly between 4 and 6. Then, between 6 and 8.

25)



H

26)



Now, by up-heap bubbling, swap operations are made between (5,12), (5,10), (5,8)

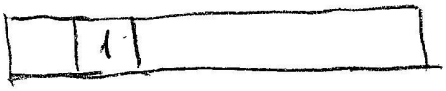
→ now 5 is at root.

At this step, heap property is satisfied, 5 is smallest key and at root.

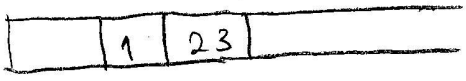
I

27) $1 \equiv 1 \bmod(11)$, note that index $= (h(x) + f(i)) \bmod 1$

where $f(i) = i^2$



$23 \equiv 2 \bmod(n)$



$45 \equiv 1 \bmod(11) \Rightarrow$ collision $\Rightarrow i=2 \Rightarrow$ index $= 1+4$

$34 \equiv 1 \mod(11) \Rightarrow i = 3 \Rightarrow index = 1 + 9 \pmod{11}$

$12 \equiv 1 \mod(11) \Rightarrow i = 4 \Rightarrow index = 1 + 16 \pmod{11}$

$= 6 \mod(11)$

is the final state.

d

28)
```
merge_sort(S1, S2, S)
    n = len(S)
    if n < 2:
        return
    mid = n // 2
    S1 = S[0:mid]
    S2 = S[mid:n]
    merge_sort(S1)
    merge_sort(S2)
    merge(S1, S2, S)
```

29) Merge sort is a better solution when we want to sort seqences whose problem size is small (50, 100 etc)

We have $1073741847 \approx 2^{30}$ numbers, if we can take them into a hash table, $2^{30}/2^{10} = 2^{20}$ numbers can be kept in same sorted or unsorted table.

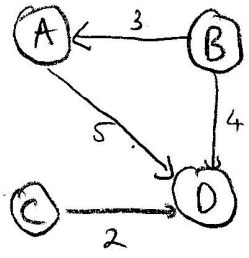
So, in this case, using <u>bucket sort</u> is better solution.

K

30) The graph is directed weighted. Let M[i][j] be the adjacency matrix of the graph.

$M[i][j] =$

	A	B	C	D
A	∞	∞	∞	5
B	3	∞	∞	4
C	∞	∞	∞	2
D	∞	∞	∞	∞



∞ : there is no edge from i to j or j to i or i to i