FINAL EXAM

Name: Ahmer Tariq

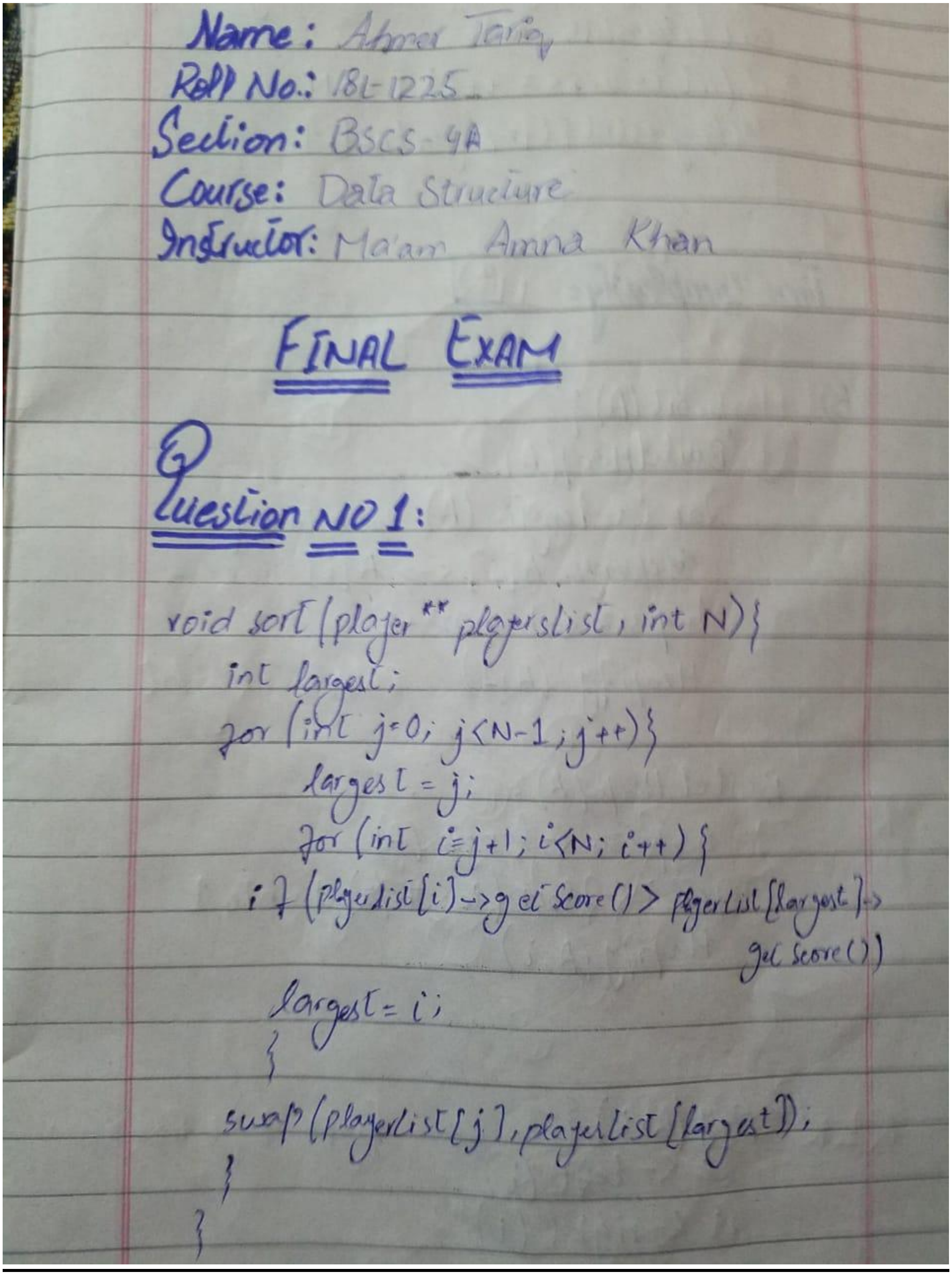
Roll No.: 18L-1225

Section: BCS(4A)

Course: Data Structure

Instructor: Ma'am Amna Khan

Question No 1:



Name: Ahmed Tariq

Roll No.: 18L-1225

Section: BSCS-4A

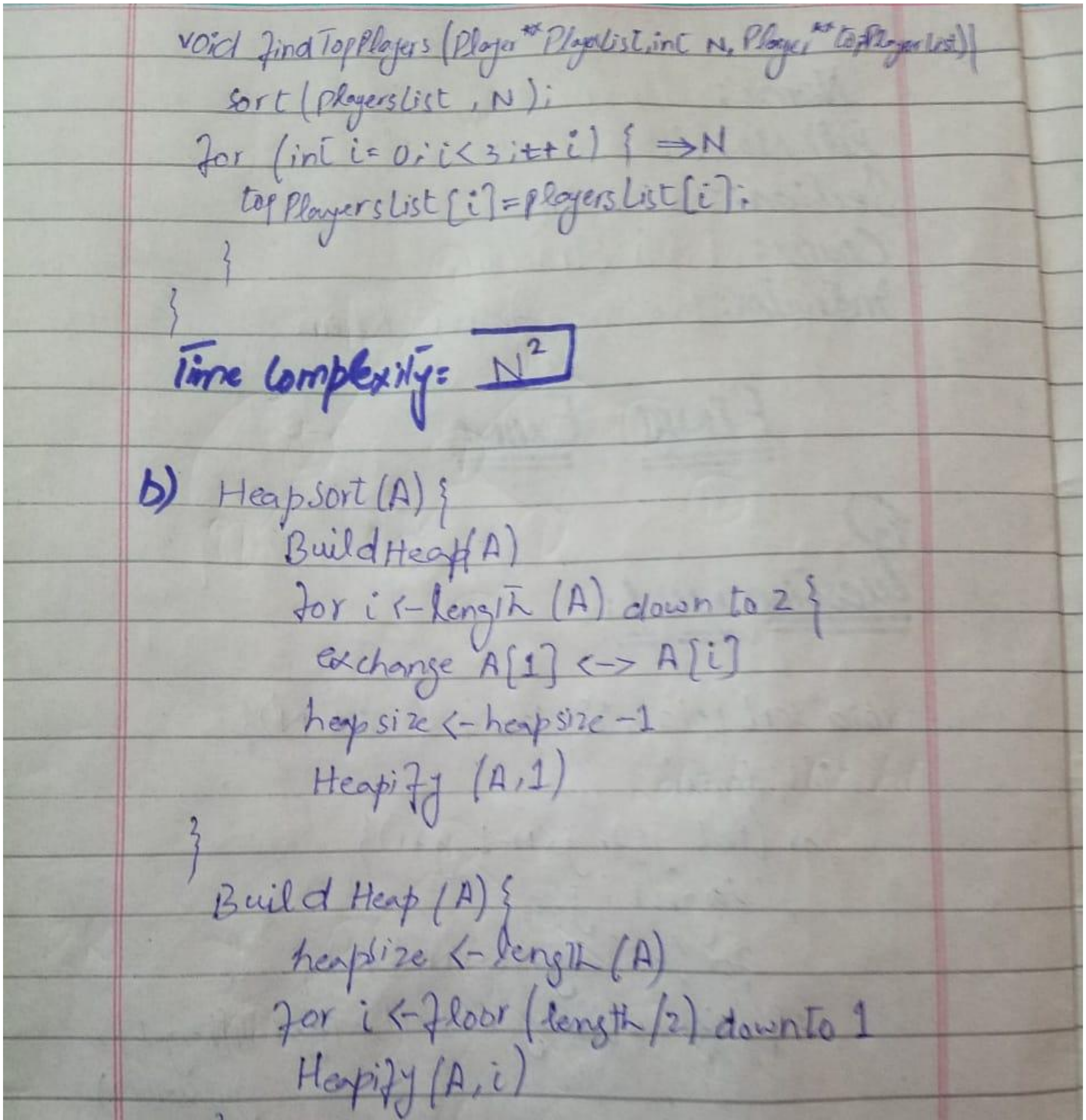
Course: Data Structure

Instructor: Ma'am Amna Khan

FINAL EXAM

Question NO 1:

```
void sort(player** playerslist, int N){
   int largest;
   for(int j=0; j<N-1; j++){
       largest = j;
       for(int i=j+1; i<N; i++){
   if(playerlist[i]->get Score() > playerList[largest]->
                                                get Score())
       largest= i;
       }
   swap(playerlist[j], playerList[largest]);
   }
}
```



```
void findTopPlayers (Player** PlayerList, int N, Player** topPlayerList) {
  sort (playersList , N);
 for (int i = 0; i < 3; ++i) { ⇒N
   topPlayersList [i] = playersList [i];
  }
}
```

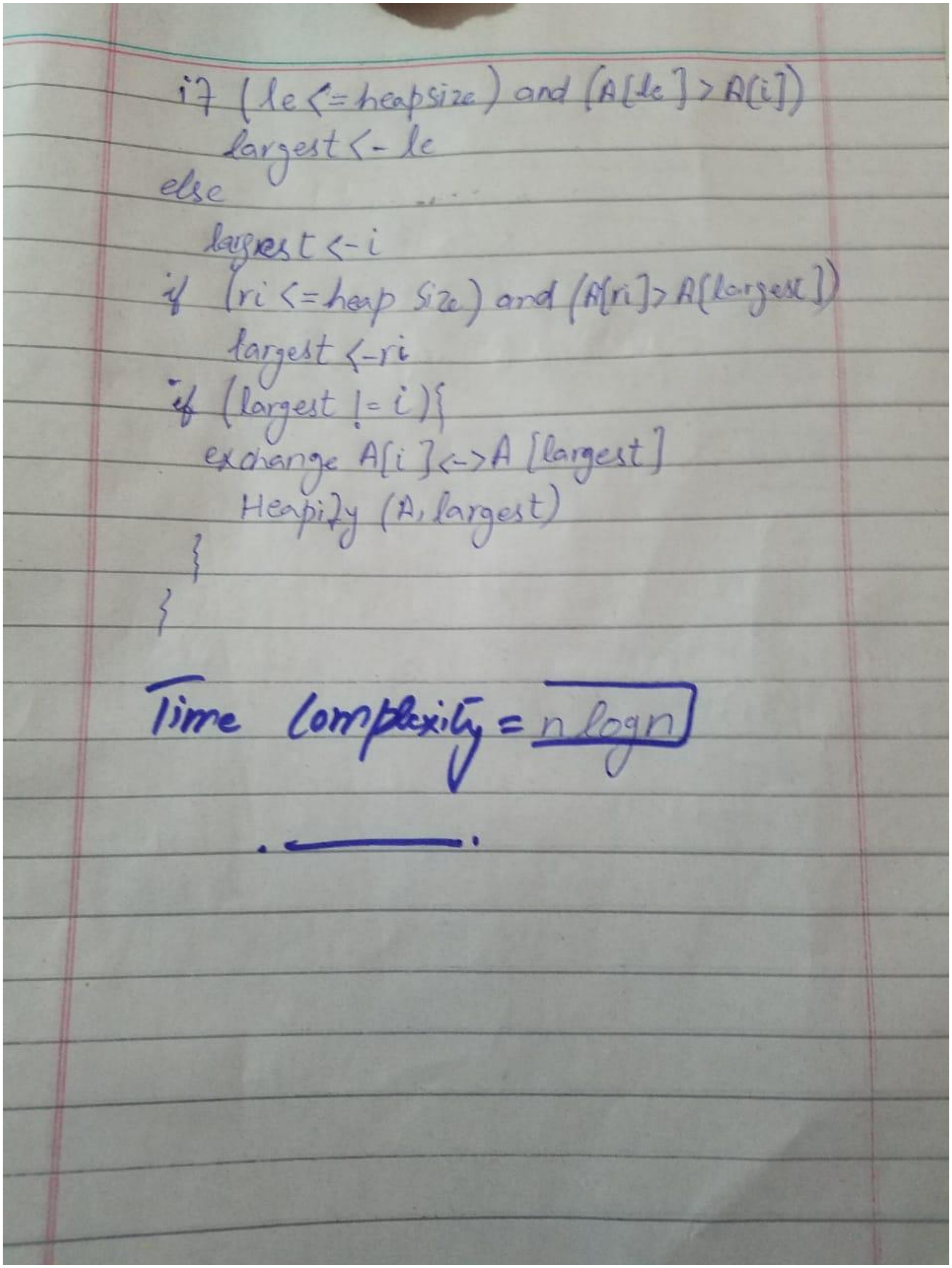
Time Complexity: $\boxed{N^2}$

b)

```
Heapsort (A) {
   BuildHeap (A)
   for i <- length (A) down to 2 {
     exchange A[1] <-> A[i]
     heapsize <- heapsize -1
     Heapify (A, 1)
}

 Build Heap (A) {
     heapsize <- length (A)
     for i <- floor (length/2) downto 1
       Heapify (A, i)
  }


Heapify (A, i) {
     le <- left (i)
     ri <- right (i)
```



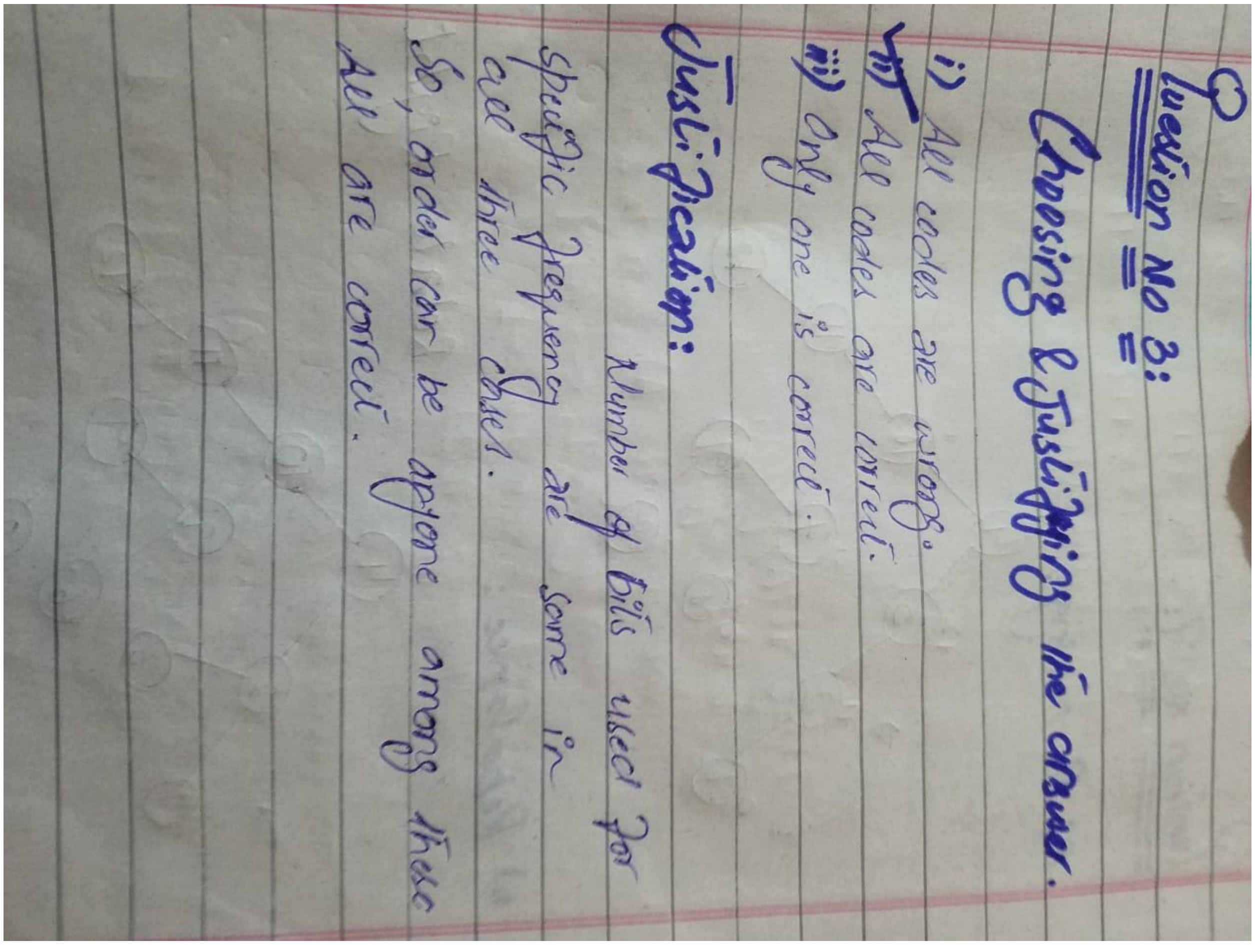
```
if (le <= heapsize) and (A[le] > A[i])
    largest <- le
else
   largest <- i
if (ri <= heap Size) and (A[ri] > A[largest])
    largest <-ri
if (largest != i){
   exchange A[i]<->A [largest]
     Heapify (A, largest)
  }
}
```

Time Complexity = n logn

Question No 2:

.cpp file attached.

Question No 3:



Question No 3:

Choosing & Justifying the answer.

i) All codes are wrong.

✓ ii) All codes are correct.

iii) Only one is correct.

Justification:

Number of bits used for specific frequency are same in all three cases. So, order can be anyone among these All are correct.

Question No 4:

Question NO 4:

a)

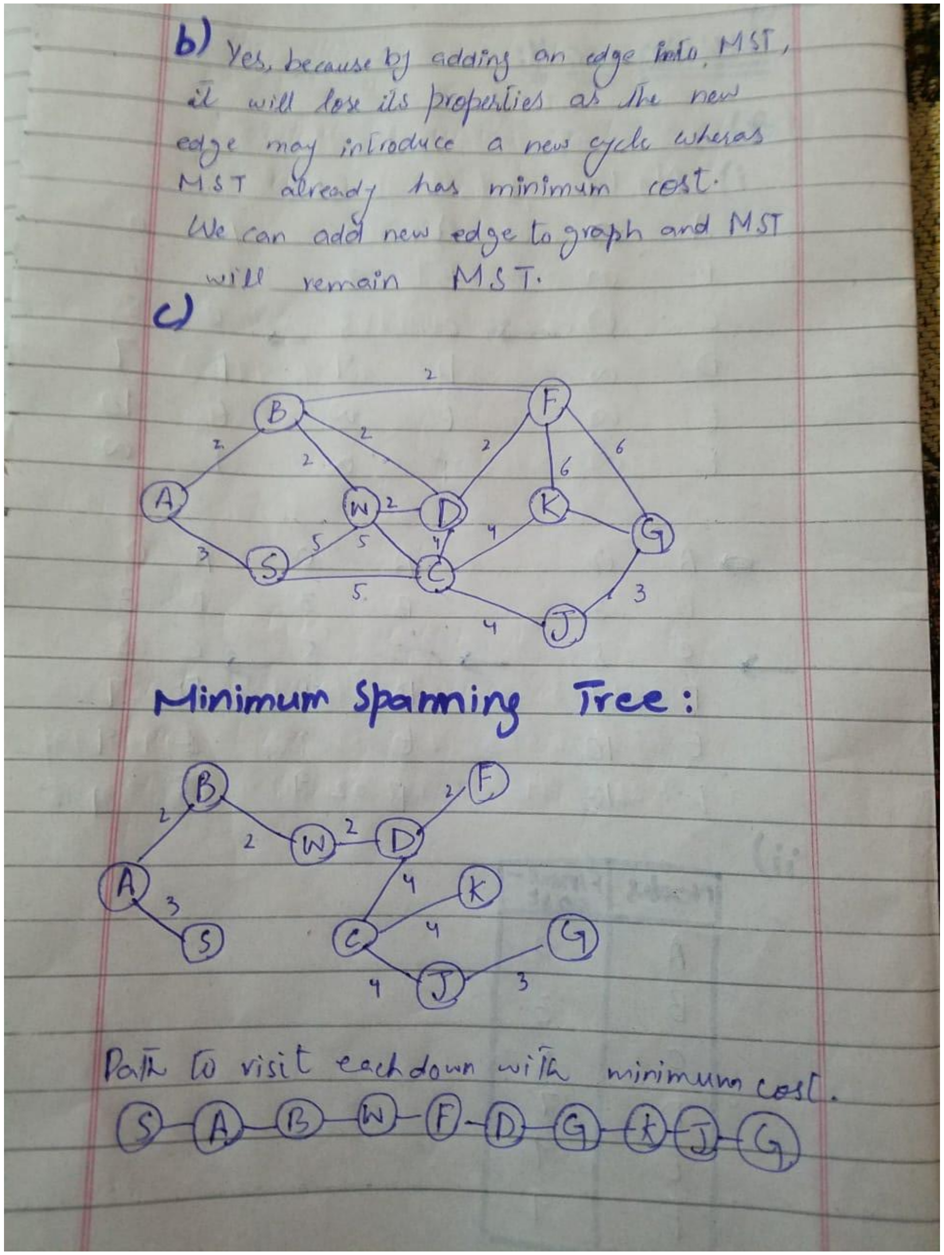
i) Nodes

A	0		A	0		A	0
B	∞		B	3		B	3
C	∞	→	C	5	→	C	5
D	∞		D	9		D	7
E	∞		E	∞		E	10
F	∞		F	∞		F	∞

	A	0		A	0		A	0
	B	3		B	3		B	3
→	C	5	→	C	5	→	C	5
	D	7		D	7		D	7
	E	10		E	9		E	9
	F	13		F	9		F	9

ii)

Nodes	Final-cost
A	0
B	3
C	5
D	7
E	9
F	9



b) Yes, because by adding an edge into MST, it will lose its properties as the new edge may introduce a new cycle whereas MST already has minimum cost.

We can add new edge to graph and MST will remain MST.

c)

A
B
F
W
D
K
G
S
C
J
2
2
2
2
2
2
6
6
3
5
5
4
4
5
4
3

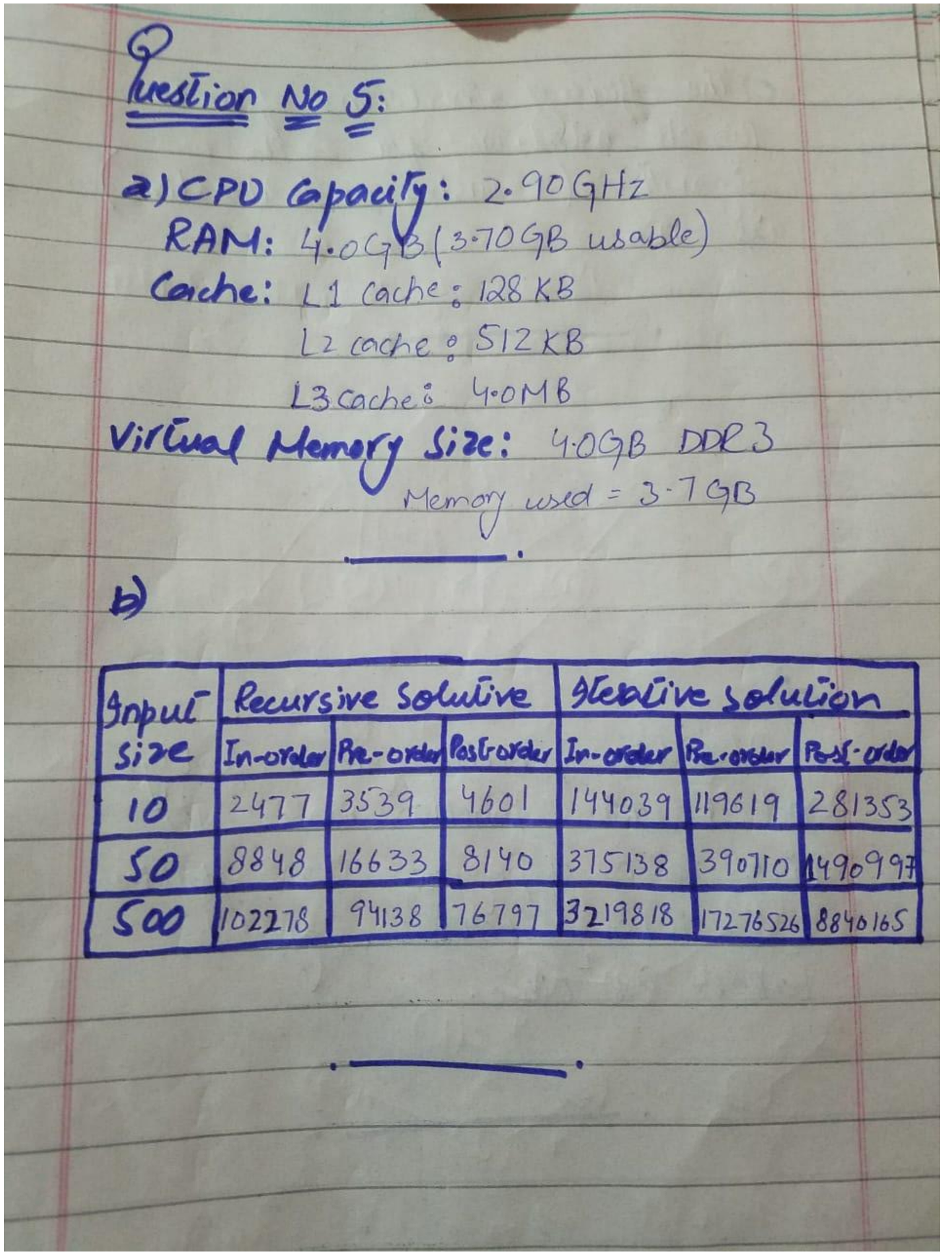
Minimum Spanning Tree :

B
F
A
W
D
K
S
C
G
J
2
2
2
2
3
4
4
4
3

Path to visit each down with minimum cost.

S — A — B — W — F — D — G — K — J — G

Question No 5:

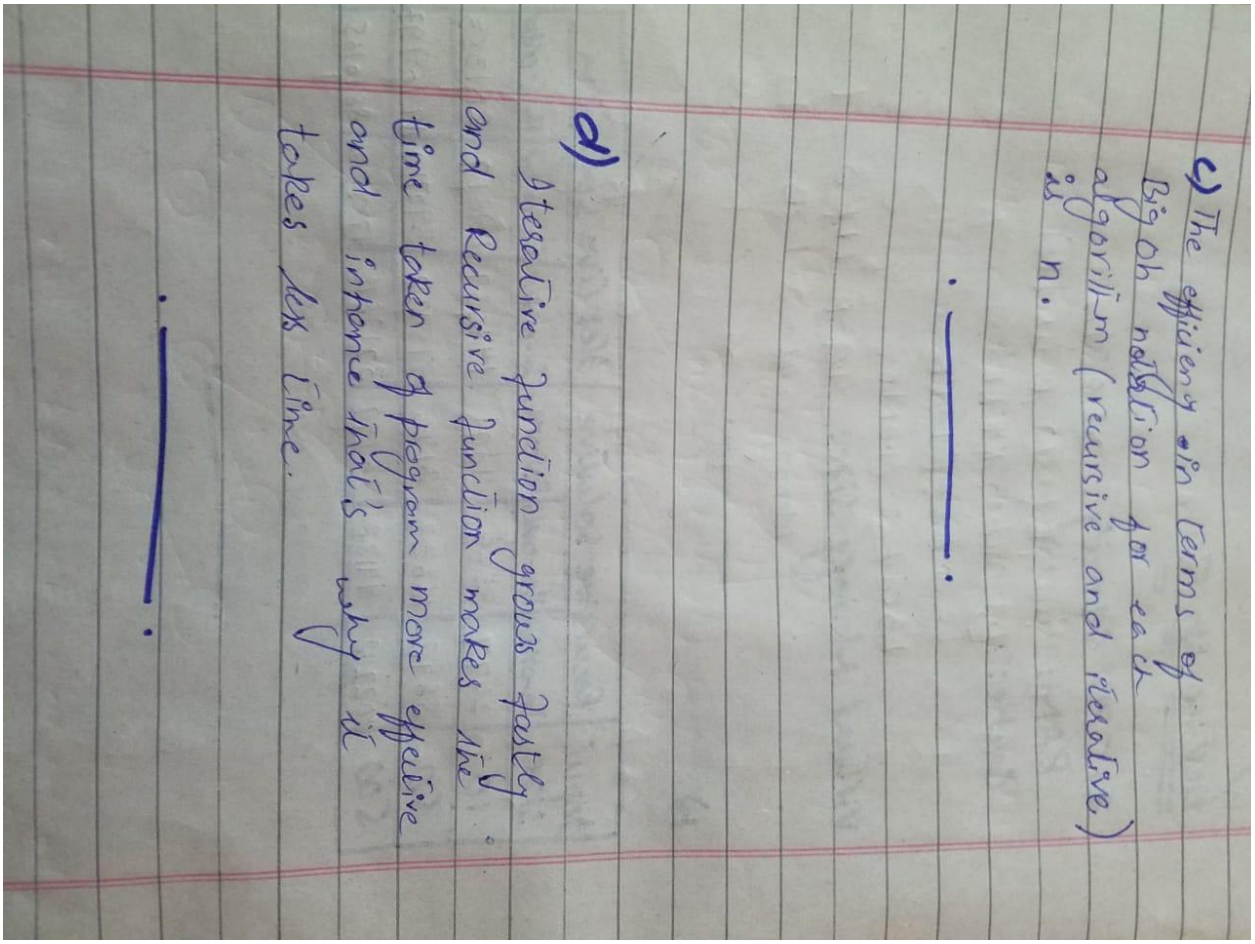


Question No 5:

a) CPU Capacity: 2.90GHz
RAM: 4.0GB (3.70GB usable)
Cache: L1 cache: 128 KB
L2 cache: 512KB
L3 cache: 4.0MB
Virtual Memory Size: 4.0GB DDR3
Memory used = 3.7GB

b)

Input size	Recursive Solutive			Iterative solution		
	In-order	Pre-order	Post-order	In-order	Pre-order	Post-order
10	2477	3539	4601	144039	119619	281353
50	8848	16633	8140	375138	390710	1490997
500	102278	94138	76797	3219818	17276526	8840165



c) The efficieny in terms of Big Oh notation for each algorithm (recursive and iterative) is n.

d) Iterative function grows fastly and Recursive function makes the time taken of program more effective and inhance that's why it takes less time.

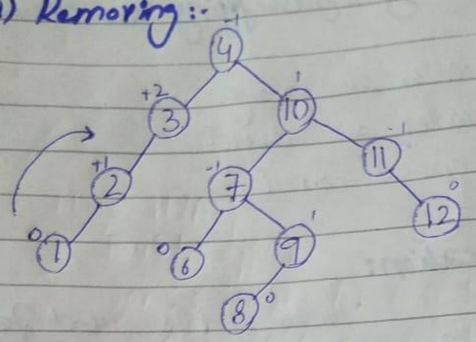
Question No 6:

.cpp file attached.

Question No 7:

Question no 7:

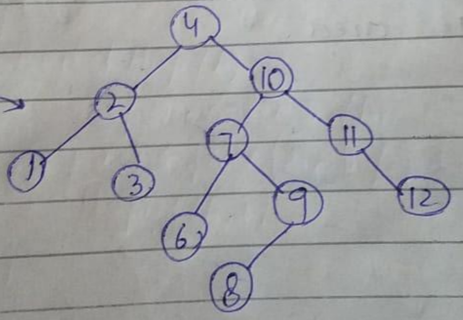
a) Removing :-



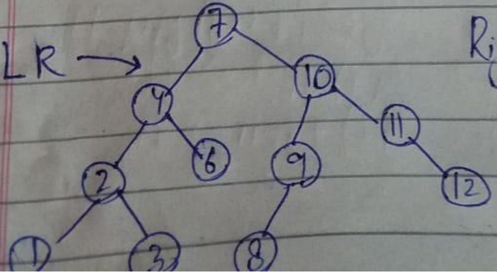
b) Rebalance :

Rotate Right

RR →



RLR → Right left Rotate.


7
4
10
2
6
9
11
1
3
8
12