Question - 1

Question 1

SCORE: **10 points**

```
abstract class A
{
    abstract int firstMethod(int i);

    abstract int secondMethod(int i);

    int thirdMethod(int i)
    {
        return secondMethod(i++);
    }
}

abstract class B extends A
{
    @Override
    int secondMethod(int i)
    {
        return firstMethod(++i);
    }
}

class C extends B
{
    @Override
    int firstMethod(int i)
    {
        return ++i;
    }
}


class HelloWorld{

    public static void main(String []args){
        C c = new C();


System.out.println(c.thirdMethod(121121));

    }
}
```

what's the output of the segment above?

- 121121
- 121122
- 121123
- 121124

Help

Question - 2
Question 2

SCORE: **10 points**

```
public class Q2 {
    public static void main(String[] args) {
        int i =111;
        System.out.print(++i+i+++" "+i);
    }
}
```

what's the output of the segment above?

- 223 112
- 225 113
- 224 113
- Compile Error

Question - 3
Question 3

SCORE: **10 points**

Which answer(s) match the given pattern?
pattern: **a*b+**

- aaaaa
- aaaaab
- a
- b
- abbb
- aaaaabbbb

Question - 4
Question 4

SCORE: **10 points**

```
public class Q4 {
    public static void main(String args[]){
        try{
            return;
        }
        finally{
            System.out.println("Quiz 3 now");
        }
    }
}
```

- Code runs with no output

- Compile time error
- Quiz 3 now
- Run time error

Question - 5

SCORE: **10 points**

Question 5

Which of the following is **FALSE** about abstract classes in Java?

If we derive an abstract class and do not implement all the abstract methods, then the derived class should also be marked as abstract using 'abstract' keyword

- Abstract classes can have constructors
- A class can be made abstract without any abstract method
- A class can inherit from multiple abstract classes.

Question - 6

SCORE: **50 points**

Coding :

Students are asked to stand in non-decreasing order of heights in line.
write a function when given an array tells how many students are not following this rule
Input: [1,1,4,2,1,3,1]
Output: 3
Explanation:
[..4, 2..] 2 is standing in front of 4 and violating the rule
[..2, 1..] 1 is standing in front of 2 and violating the rule
[..3, 1] 1 is standing in front of 3 and violating the rule
input:- the array of student heights
output:- (return type int)
number of students not following the rule
Note:
•) 1 <= heights.length <= 100
•) 1 <= heights[i] <= 10
•) The first person in the line is considered to always follow the rule as there is no one in front

If you are going to test your case, the input(array)'s form is:
1 2 3 4 5 6 7 8
(Splited by blank