Sample exam 3

The INFDEV team

1 Question 1

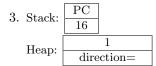
Given the following block of code, fill in the stack, heap, and PC with all the steps taken by the program at runtime.

- Points: 4 (50% of total).
- Grading: one point per correctly filled-in execution step.
- Associated learning objective: abstraction.

```
interface MovableObject {
   2
      class Car : MovableObject {
        private float direction = 3.14;
   5
        public Car() {
   6
        }
        static public void move(Car car,float direction) {
   8
          car.direction = direction;
   9
        }
  10
      class Particle : MovableObject {
  11
  12
        private float direction = 0;
  13
        public Particle() {
  14
  15
∾ <sub>16</sub>
     MovableObject mo = new Car();
  17
      Car.move(mo,1);
```

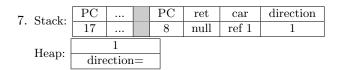




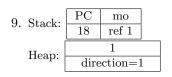




5.	Stack:	PC			PC	ret
		16			6	ref 1
	Heap:	1				
		direction=				







2 Question 2

Given the following block of code, fill in the declarations, class definitions, and PC with all steps taken by the compiler while type checking.

- Points: 4 (50% of total).
- Grading: one point per correctly filled-in type checking step.
- \bullet Associated learning objective: $type\ checking.$

```
interface IntList {
2
     bool isEmpty();
3
     int getValue();
4
5
    class IntNode : IntList {
6
     private int value;
     private IntList tail;
     public IntNode(int value,IntList tail) {
9
       this.value = value;
10
       this.tail = tail;
11
     }
12
     public bool isEmpty() {
13
       return false;
14
     }
     public int getValue() {
15
16
        return this.value;
17
     }
```

```
18
   class IntEmpty : IntList {
19
20
     public IntEmpty() {
21
     public bool isEmpty() {
22
23
       return true;
24
     }
25
     public int getValue() {
26
       return 0;
27
     }
28
29
   IntList list = new IntNode(5,new IntEmpty());
```

1. Declarations: $\begin{array}{|c|c|c|}\hline PC \\ \hline 1 \\ \hline \end{array}$

2. Declarations: PC 5

3. Declarations: PC 18

Classes:

4. Declarations: $\begin{array}{|c|c|c|}\hline PC \\ \hline 28 \\ \hline \end{array}$

	IntEmpty	IntList	IntNode	
Classes:			$IntNode = (IntNode \times int \times IntList) \rightarrow IntNode$	
	$IntEmpty = IntEmpty \rightarrow IntEmpty$	$getValue=getValue \rightarrow int$	$getValue = getValue \rightarrow int$	
	$getValue = getValue \rightarrow int$	$\begin{array}{c} \text{get varue} = \text{get varue} \rightarrow \text{int} \\ \text{isEmpty} = \text{isEmpty} \rightarrow \text{bool} \end{array}$	$isEmpty=isEmpty \rightarrow bool$	
	$isEmpty=isEmpty \rightarrow bool$	$ $ is Empty = is Empty \rightarrow boof	tail=IntList	
			value=int	

5. Declarations: $\begin{array}{|c|c|c|}\hline PC \\\hline 30 \\\hline \end{array}$

	IntEmpty	IntList	IntNode	
Classes:	Interests Interests Interests		$IntNode = (IntNode \times int \times IntList) \rightarrow IntNode$	
	$IntEmpty=IntEmpty \rightarrow IntEmpty \\ getValue=getValue \rightarrow int$	$getValue = getValue \rightarrow int$ isEmpty=isEmpty \rightarrow bool	$getValue=getValue \rightarrow int$ $isEmpty=isEmpty \rightarrow bool$	
	$isEmpty=isEmpty \rightarrow bool$	$isEmpty=isEmpty \rightarrow bool$	tail=IntList	
			value=int	