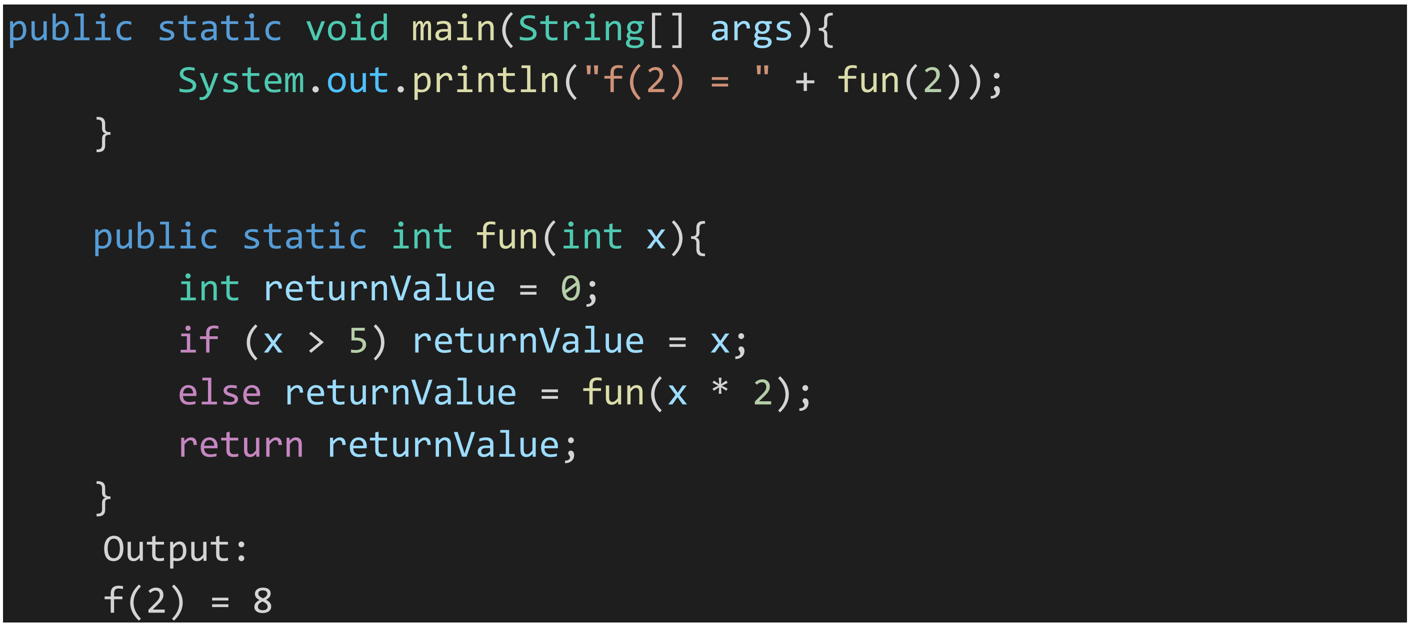
Question 1:

```
import java.util.Scanner;
public class Test {
    public static void main(String[] args){
        System.out.print("Enter the list size: ");
        int size = new Scanner(System.in).nextInt();

        System.out.println("Enter " + size + " numbers:");
        int[] arr = new int[size];
        for (int i = 0; i < size; i++){
            arr[i] = new Scanner(System.in).nextInt();
        }

        countOccurrence(arr);
    }

    public static void countOccurrence(int[] array){
        int[] frequency = new int[10];
        for (int i : array){
            frequency[i]++;
        }
        for (int i = 0; i < 10; i++){
            if(frequency[i] == 0) continue;
            else if (frequency[i] == 1)
                System.out.println(i + " Occurres " +
frequency[i] + " time");
            else
                System.out.println(i + " Occurres " +
frequency[i] + " times");
        }
    }
}
```

Question 2:

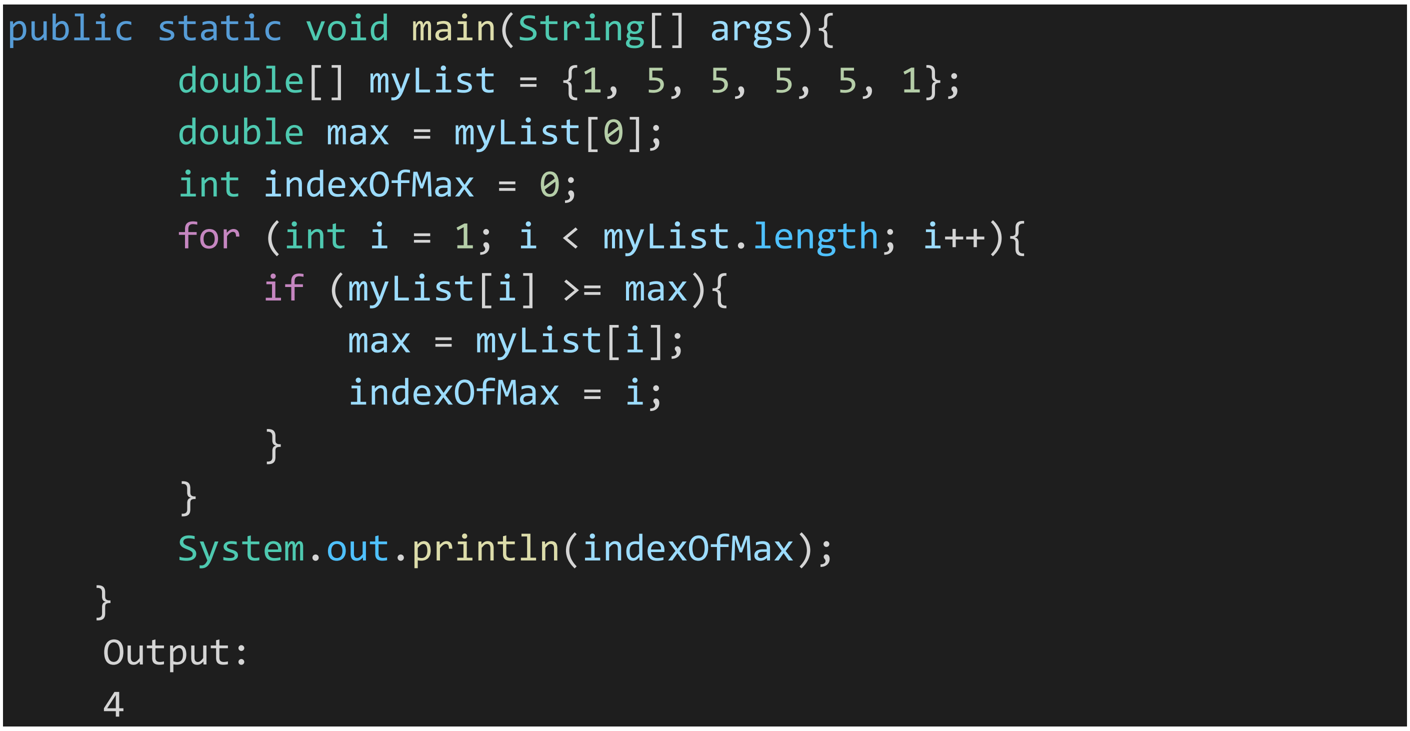
```
public class InvoiceItem {
    private String id;
    private String desc;
    private int qty;
    private double unitPrice;

    // Constructors
    public InvoiceItem(String i, String d, int q, double u){
        id = i;
        desc = d;
        if (q > 0) qty = q;
        else System.out.println("QTY can\'t be zero or negative
value");
        if (u > 0) unitPrice = u;
        else System.out.println("UnitPrice can\'t be zero or
negative value");
    }

    // Setters
    public void setQTY(int q){
        if (q > 0) qty = q;
        else System.out.println("QTY can\'t be zero or negative
value");
    }

    public void setUnitPrice(double u){
        if (u > 0) unitPrice = u;
        else System.out.println("UnitPrice can\'t be zero or
negative value");
    }

    // Getters
    public String getID(){
        return id;
    }
    public String getDesc(){
        return desc;
    }

    public int getQTY(){
        return qty;
    }

    public double getUnitePrice(){
        return unitPrice;
    }

    public double getTotal(){
        return qty * unitPrice;
    }

    // Other Methods
    public String toString(){
        String message = "ID : " + id + "\nDesc : " + desc +
"\nUnitePrice : " + unitPrice + "\nQTY : " + qty + "\nTotal : " +
this.getTotal();
        System.out.println(message);
        return message;
    }

    // Main
    public static void main(String[] args){
        InvoiceItem Ahmed = new InvoiceItem("101", "Food", 3,
2.5);
        String temp = Ahmed.toString();
    }
}
    Output:
    ID : 101
    Desc : Food
    UnitePrice : 2.5
    QTY : 3
    Total : 7.5
```

Question 3[1]:



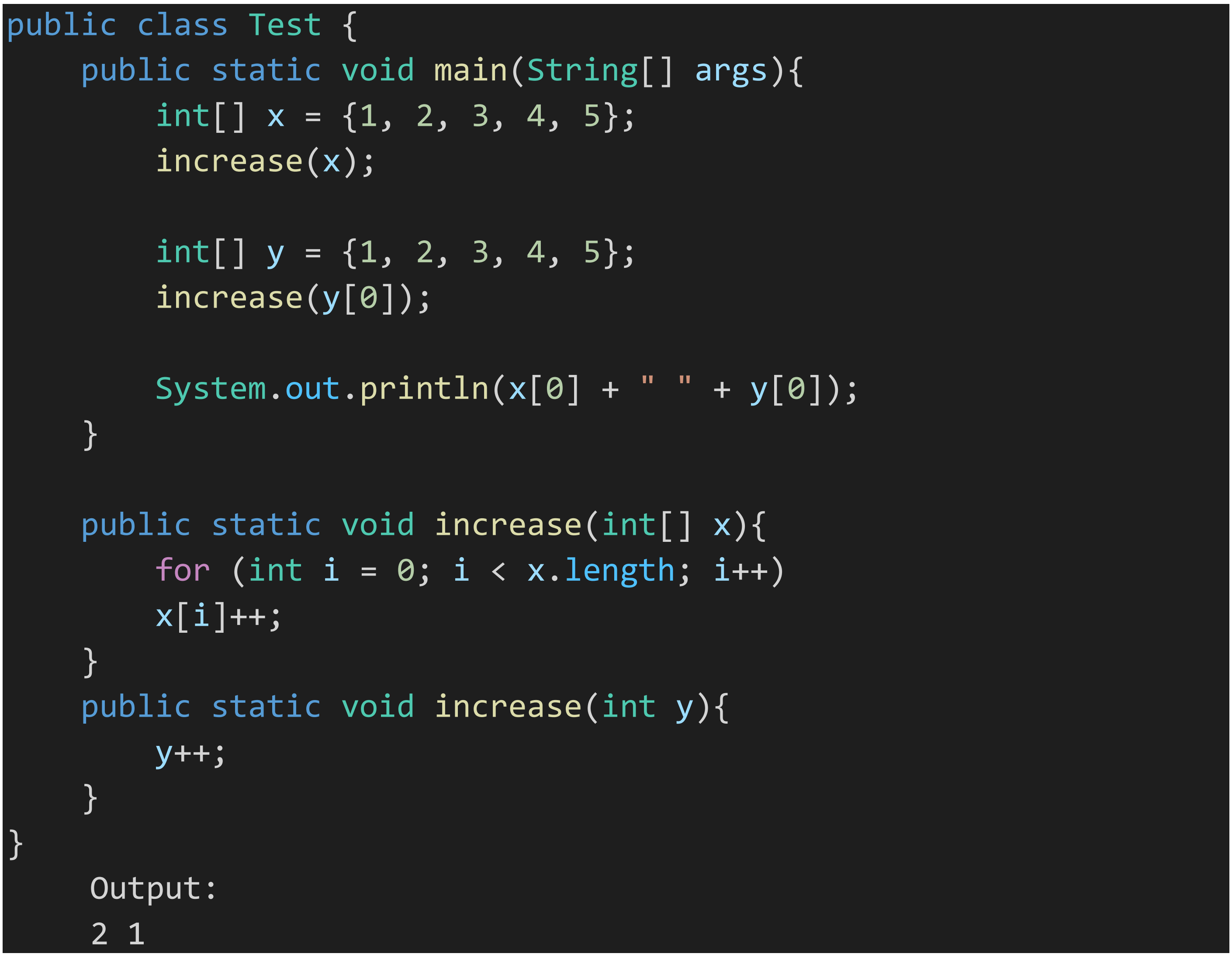
```
public static void main(String[] args){
        System.out.println("f(2) = " + fun(2));
    }

    public static int fun(int x){
        int returnValue = 0;
        if (x > 5) returnValue = x;
        else returnValue = fun(x * 2);
        return returnValue;
    }
     Output:
     f(2) = 8
```

Question 3[2]:



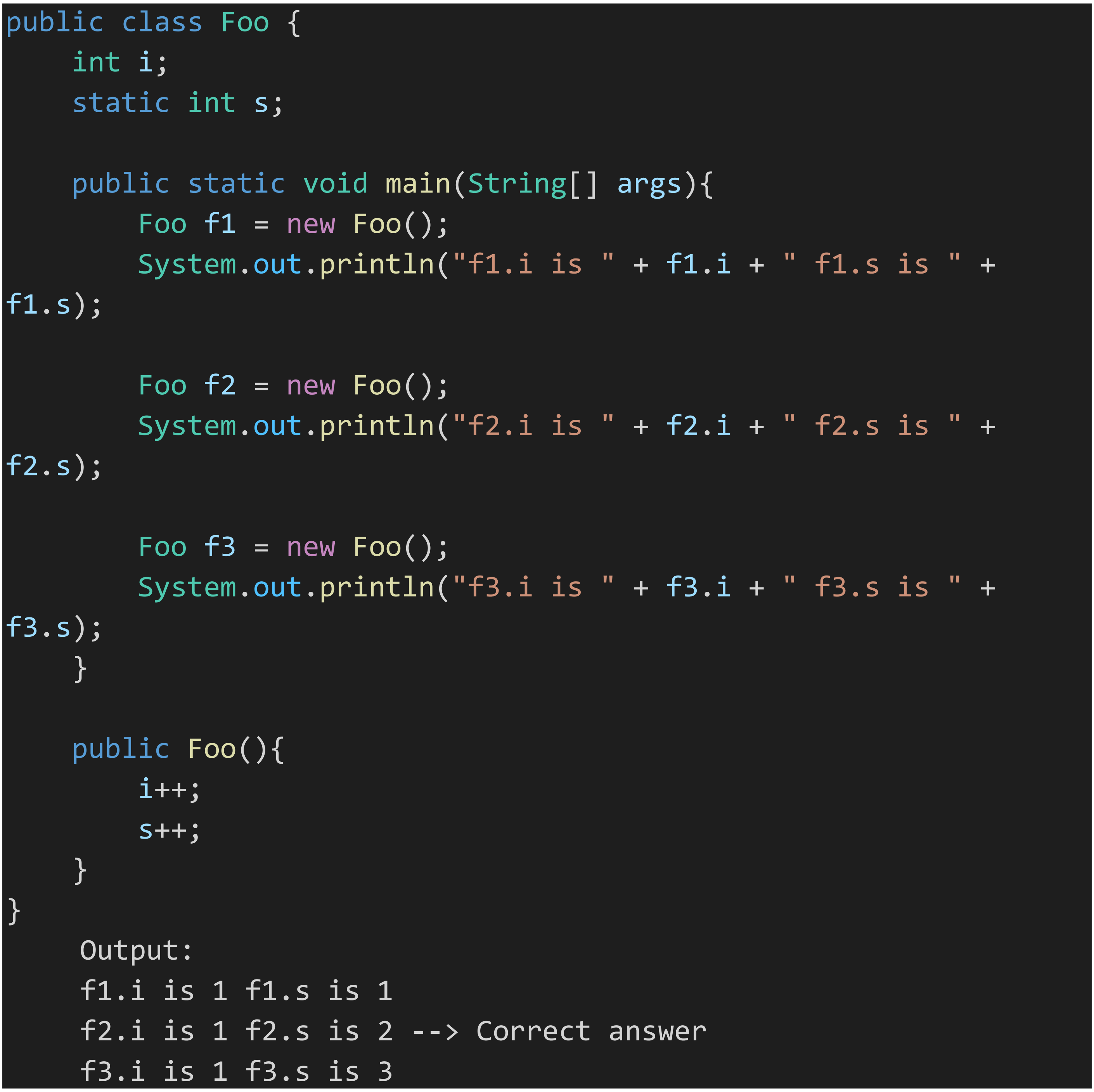
```
public static void main(String[] args){
        double[] myList = {1, 5, 5, 5, 5, 1};
        double max = myList[0];
        int indexOfMax = 0;
        for (int i = 1; i < myList.length; i++){
            if (myList[i] >= max){
                max = myList[i];
                indexOfMax = i;
            }
        }
        System.out.println(indexOfMax);
    }
     Output:
     4
```

Question 3[3]:



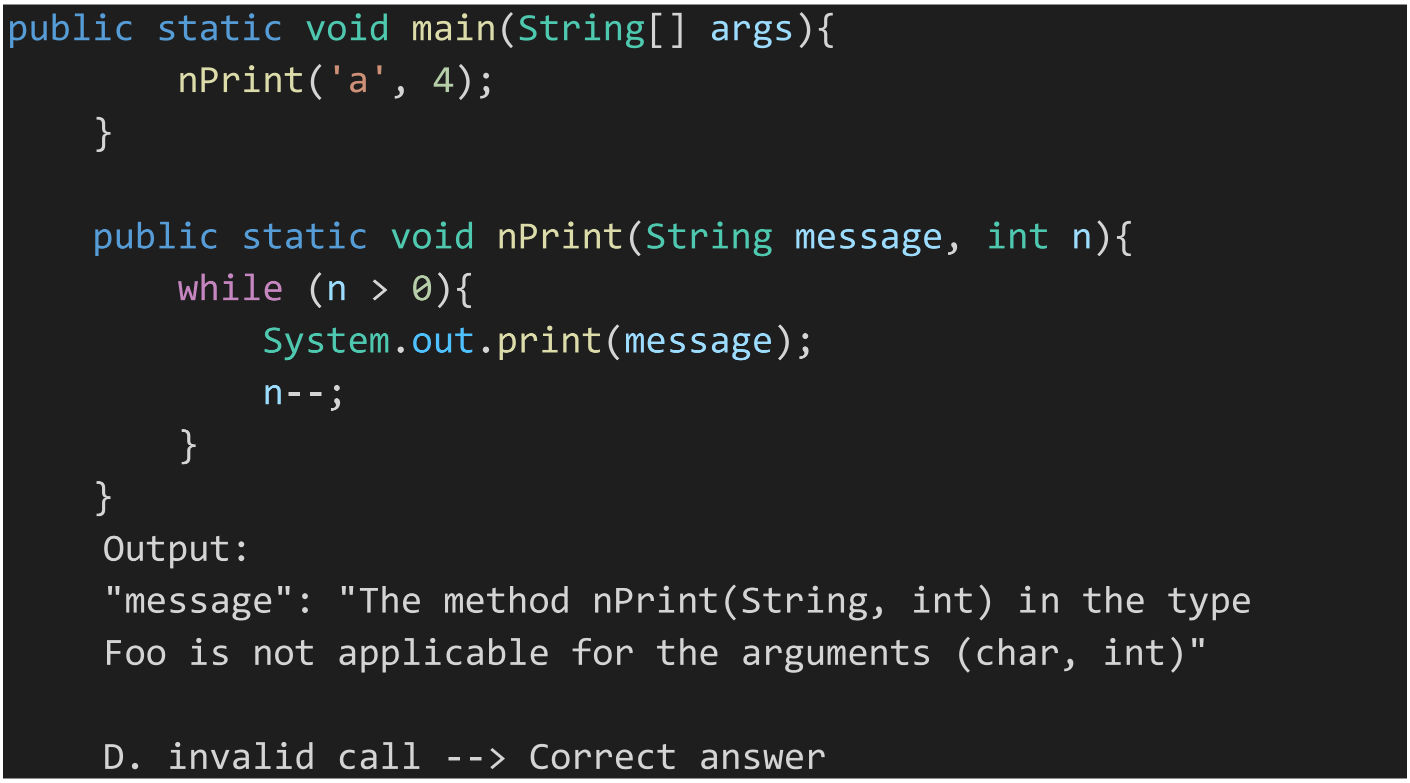
```
public class Test {
    public static void main(String[] args){
        int[] x = {1, 2, 3, 4, 5};
        increase(x);

        int[] y = {1, 2, 3, 4, 5};
        increase(y[0]);

        System.out.println(x[0] + " " + y[0]);
    }

    public static void increase(int[] x){
        for (int i = 0; i < x.length; i++)
        x[i]++;
    }
    public static void increase(int y){
        y++;
    }
}
    Output:
    2 1
```

Question 3[4]:



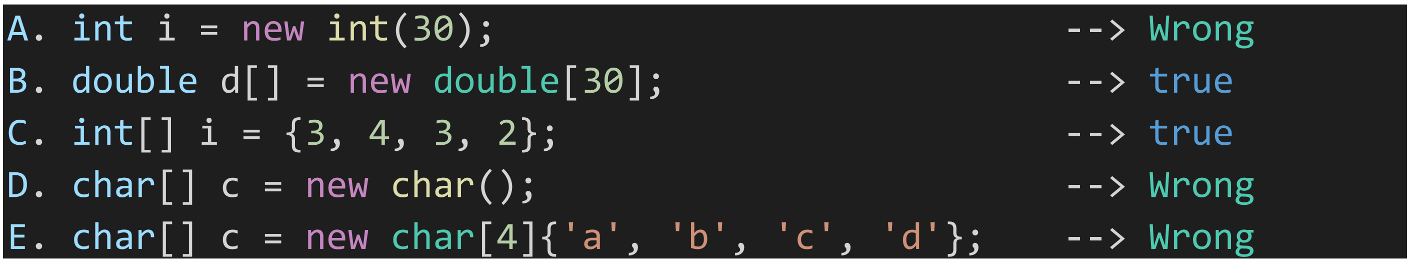
```
public class Foo {
    int i;
    static int s;

    public static void main(String[] args){
        Foo f1 = new Foo();
        System.out.println("f1.i is " + f1.i + " f1.s is " +
f1.s);

        Foo f2 = new Foo();
        System.out.println("f2.i is " + f2.i + " f2.s is " +
f2.s);

        Foo f3 = new Foo();
        System.out.println("f3.i is " + f3.i + " f3.s is " +
f3.s);
    }

    public Foo(){
        i++;
        s++;
    }
}
    Output:
    f1.i is 1 f1.s is 1
    f2.i is 1 f2.s is 2 --> Correct answer
    f3.i is 1 f3.s is 3
```

Question 4[1]:



```
public static void main(String[] args){
        nPrint('a', 4);
    }

    public static void nPrint(String message, int n){
        while (n > 0){
            System.out.print(message);
            n--;
        }
    }
    Output:
    "message": "The method nPrint(String, int) in the type
    Foo is not applicable for the arguments (char, int)"

    D. invalid call --> Correct answer
```

Question 4[2]:



```
A. int i = new int(30);                           --> Wrong
B. double d[] = new double[30];                   --> true
C. int[] i = {3, 4, 3, 2};                        --> true
D. char[] c = new char();                         --> Wrong
E. char[] c = new char[4]{'a', 'b', 'c', 'd'};    --> Wrong
```