Question - 1

Binary Search

SCORE: **5 points**

Given a sorted integer array of length N, what's the maximum number of comparisons required to find a target number using the binary search method ?

- ( ) logN
- ( ) log(N+1)
- (•) logN +1
- ( ) logN -1

Question - 2

Complexity of the Code Snippet

SCORE: **5 points**

Core CS | Algorithms | Complexity | Easy

Consider the following code snippet:

```
int a = 1;

while (a < n) {
    a = a * 2;
}
```

What is the complexity of the above code snippet?

- ( ) O(n)
- ( ) O(1)
- (•) $O(\log_2(n))$
- ( ) $O(2^n)$

Question - 3

Implementation

SCORE: **20 points**

Your company has many different products and each products comes in different sizes.
The inventory management system treats each combination of "(*String*) *productName*" & "(*int*) *size*" as an individual *Product* record.

Help

The records also contains a "(*LocalDateTime*) *lastUpdate*" field as a timestamp, which can be updated so that, even if it is different, as long as the *productName* and *size* fields are equal, the *Product* records should be considered the same.

> product1: productName = "Gloves", size = 5, timestamp = "2018-01-15 18:35:05"
> product2: productName = "Gloves", size = 7, timestamp = "2018-01-11 10:02:55"
> product3: productName = "Gloves", size = 5, timestamp = "2017-12-03 11:28:23"
> product4: productName = "Hat", size = 5, timestamp = "2017-12-03 11:28:23"

Only *product1.equals(product3)* should return *true* as their *productName* and *size* fields are identical (the timestamp may vary.)
All other *equals()* calls should return false.

Please complete both *equals()* and *hashCode()* methods for the *Product* class.

Question - 4

SCORE: **15 points**

Pass by value or reference

Problem Statement

When the following code is executed, there will be 6 lines of output generated. Please complete the output.

```
public class CallByValue {
    int number = 0;
    int[] array = {0};

    public int incrementNumber1(int number) {
        number++;
        return number;
    }

    public int incrementNumber2() {
        number++;
        return number;
    }

    public int[] incrementArray1 (int[] array) {
        for (int i = 0; i < array.length; i++) {
            array[i]++;
        }
        return array;
    }

    public int[] incrementArray2 () {
        for (int i = 0; i < array.length; i++) {
            array[i]++;
        }
        return array;
    }

    public static void main(String[] args) {
        CallByValue cbv = new CallByValue();

        cbv.number = 0;
        cbv.incrementNumber1(cbv.number);
        System.out.println("incrementNumber1(): "
+ cbv.number);
```

```
        cbv.number = 0;
        cbv.incrementNumber2();
        System.out.println("incrementNumber2(): "
+ cbv.number);

        cbv.array[0] = 0;
        cbv.incrementArray1(cbv.array);
        System.out.println("incrementArray1(): "
+ cbv.array[0]);

        cbv.array[0] = 0;
        cbv.incrementArray2();
        System.out.println("incrementArray2(): "
+ cbv.array[0]);
    }
}
```

incrementNumber1(): <blank 1> incrementNumber2(): <blank 2>
incrementArray1(): <blank 3> incrementArray2(): <blank 4>

Answers

<blank 1> : [0]
<blank 2> : [1]
<blank 3> : [1]
<blank 4> : [1]

Question - 5

SCORE: **5 points**

What is the result of this code?

```
import java.util.*;

public class VLA2 {
    public VLA2(int d) { dishSize = d; }

    public static void main(String[] args) {
        Comparator<VLA2> cf = new
Comparator<VLA2>() {
            @Override
            public int compare(VLA2 o1, VLA2 o2)
{
                return
Integer.compare(o2.dishSize,o1.dishSize);
            }
        };
        VLA2[] va = {new VLA2(40), new VLA2(200),
new VLA2(60)};
        Arrays.sort(va, cf);
        int index = Arrays.binarySearch(va, new
VLA2(40), cf);
        System.out.print(index + " ");
        index = Arrays.binarySearch(va, new
VLA2(80), cf);
        System.out.print(index>=0);
    }

    private int dishSize;
}
```

- 0 true
- 0 false
- 2 true
- 2 false
- Compilation Error
- Run time Exception