JAVA (Basics and Fundamentals) Assignment AU

In-Slide Questions

Question 1: Write a Java program to print the result of the following operations.

a) -5 + 8 * 6

b) (55+9) % 9

c) 20 + -3*5 / 8

d) 5 + 15 /3*2-8 % 3

Solution:

```
import java.util.*;
//======================== ACCOLITE UNIVERSITY JAVA ASSIGNMENT==========================
//-------------------------------------------------------------------------------------
//======================== VISHNU SINGH SENGAR==========================================
//-------------------------------------------------------------------------------------

public class Question1
{
    public static void main(String[] args)
    {
        Scanner inp = new Scanner(System.in);
        System.out.println("Enter the number of Expressions");
        int n= inp.nextInt();
        for(int co=0;co<n;co++) {
            Stack<Integer> op  = new Stack<Integer>();
            Stack<Double> val = new Stack<Double>();
            Stack<Integer> optmp  = new Stack<Integer>();
            Stack<Double> valtmp = new Stack<Double>();
            System.out.println("\nEnter expression\n");
            String input = inp.next();
            input = "0" + input;
            input = input.replaceAll("-","+-");
            String temp = "";
            for (int i = 0;i < input.length();i++)
            {
                char ch = input.charAt(i);
                if (ch == '-')
                    temp = "-" + temp;
                else if (ch != '+' &&  ch != '*' && ch != '/')
                   temp = temp + ch;
                else
                {
                    val.push(Double.parseDouble(temp));
                    op.push((int)ch);
                    temp = "";
                }
            }
            val.push(Double.parseDouble(temp));
            char operators[] = {'/','*','+'};
```

Figure 1 Code Snippet for Question1 (in slide question)

Question 2: Write a Java program to print an American flag on the screen.

Solution:

American flag can be broken into two patterns, one which contain repeating star pattern and other that contains repeating line pattern. When printed the right number of times, an American flag is rendered on the console output.

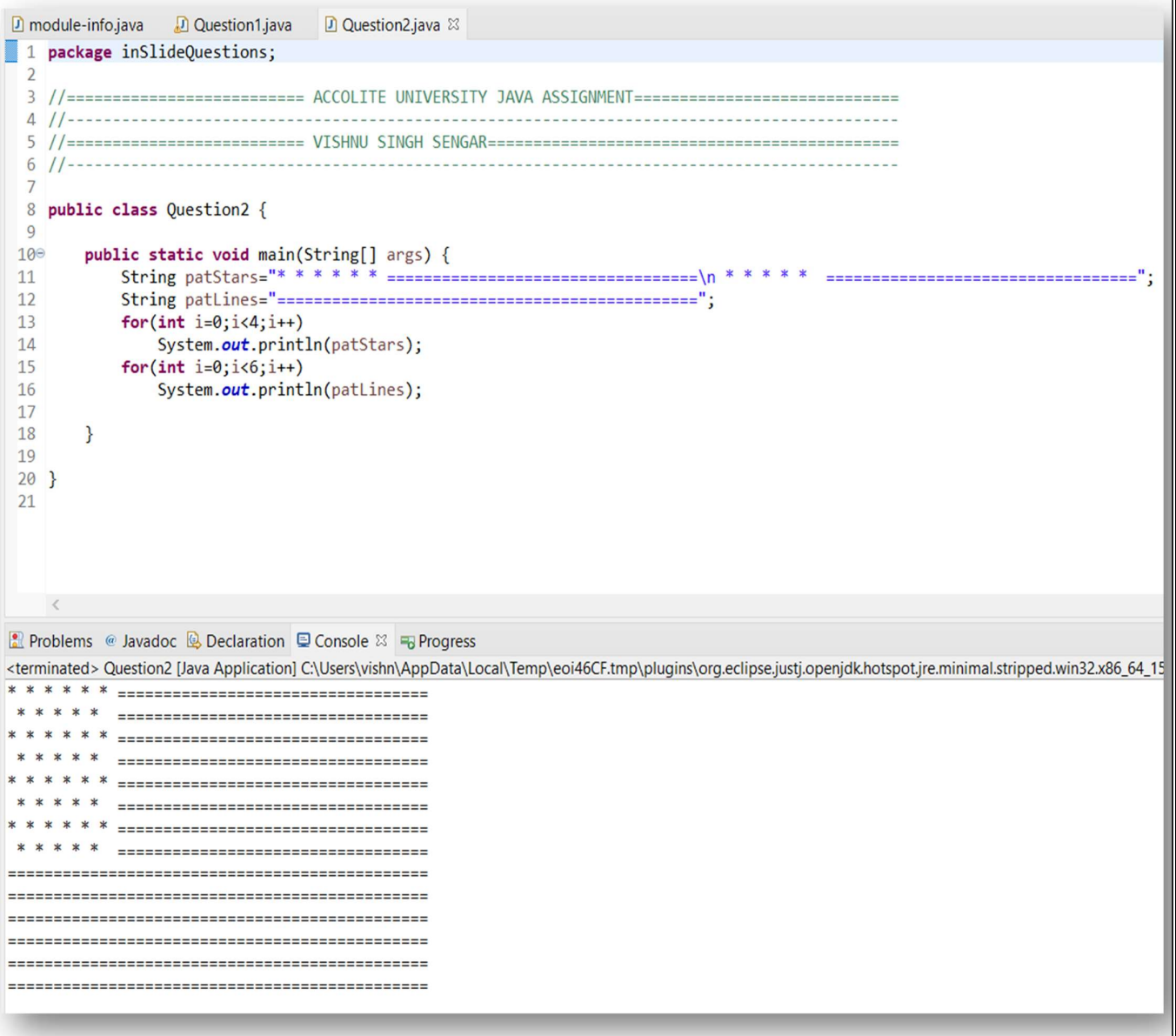

Figure 2 Code and Output for Question 2 (In Slide Assignment)

Hands On Questions

Question 1: **String**

Write a Java program to replace each substring of a given string that matches the given regular expression with the given replacement.

module-info.java | Question1.java

```
1  package handsOnQuestions;
2  import java.util.*;
3
4  public class Question1 {
5
6      public static void main(String[] args) {
7          Scanner inp=new Scanner(System.in);
8
9          System.out.println("Enter the original string: ");
10         String oriString=inp.nextLine();
11
12         System.out.println("Enter the word to be replaced: ");
13         String regEx=inp.nextLine();
14
15         System.out.println("Enter the replacement word: ");
16         String tarWord=inp.nextLine();
17
18         String newStr=oriString.replaceAll(regEx, tarWord);
19
20         System.out.println("Original String: "+oriString);
21         System.out.println("New String: "+newStr);
22
23     }
24
25 }
```

Problems | Javadoc | Declaration | Console | Progress

```
<terminated> Question1 (1) [Java Application] C:\Users\vishn\AppData\Local\Temp\eoi46CF.tmp\plugins\org.eclipse.justj.openjdk.hotspot.jre
Enter the original string:
Accolite is leading company in service sector. All clients love Accolite.
Enter the word to be replaced:
Accolite
Enter the replacement word:
Accolite Digital
Original String: Accolite is leading company in service sector. All clients love Accolite.
New String: Accolite Digital is leading company in service sector. All clients love Accolite Digital.
```

Figure 3 Code + Output for Question 1 (Hands On)

Question 2: **Collection**

Write a Java program to get a reverse order view of the keys contained in a given map.

Solution:

module-info.java | Question2.java | Question1.java

```
1  package handsOnQuestions;
2  import java.util.*;
3
4  public class Question2 {
5
6      public static void main(String[] args) {
7          TreeMap<String,String> dict= new TreeMap<>();
8          Scanner inp=new Scanner(System.in);
9
10         System.out.println("Enter the number of key->value pairs");
11         int n=inp.nextInt();
12
13         for(int i=0;i<n;i++)
14         {
15             System.out.println("Enter Key");
16             String key=inp.next();
17             System.out.println("Enter Value");
18             String val=inp.next();
19             dict.put(key, val);
20         }
21         System.out.println("Original Map: "+dict);
22         System.out.println("Reversed Map: "+ dict.descendingKeySet());
23     }
24
25
26 }
```

Console

```
<terminated> Question2 (1) [Java Application] C:\Users\vishn\AppData\Local\Temp\eoi46CF.tmp\plugins\org.eclipse.j
Enter the number of key->value pairs
5
Enter Key
Morgan
Enter Value
Accolite
Enter Key
FedEx
Enter Value
Accolite
Enter Key
MountainCapital
Enter Value
Accolite
Enter Key
Delloitte
Enter Value
Accolite
Enter Key
Wave
Enter Value
Accolite
Original Map: {Delloitte=Accolite, FedEx=Accolite, Morgan=Accolite, MountainCapital=A
Reversed Map: [Wave, MountainCapital, Morgan, FedEx, Delloitte]
```

Figure 4 Code + Output for Question 2 (HandsOn)

Question 3: **Exception**

Write your own unchecked Exception and throw it from you counter programme which counts 1 to 100.When you get Prime no while counting then throw this Exception and catch this to print you exception message.

Solution:

module-info.java | Question2.java | Question1.java | Question3.java

```
1  package handsOnQuestions;
2  import java.util.*;
3
4  class ExceptPrime extends Exception{
5
6  }
7
8  public class Question3 {
9      public static boolean checkPrime(int n) {
10         if(n<=1)
11             return false;
12         if(n<=3)
13             return true;
14         if(n%2==0||n%3==0)
15             return false;
16         for(int i=5;i*i<=n;i=i+6)
17             if(n%i==0 || n%(i+2)==0)
18                 return false;
19         return true;
20     }
21
22     public static void main(String[] args) {
23         int count=1;
24         while(count<=100) {
25             try {
26                 if(checkPrime(count))
27                     throw new ExceptPrime();
28             }
29             catch(ExceptPrime e) {
30                 System.out.println(count+" is a Prime Number");
31             }
32             finally {
33                 count++;
34             }
35         }
36     }
37
38
39 }
```

Console

```
<terminated> Question3 [Java
2 is a Prime Number
3 is a Prime Number
5 is a Prime Number
7 is a Prime Number
11 is a Prime Number
13 is a Prime Number
17 is a Prime Number
19 is a Prime Number
23 is a Prime Number
29 is a Prime Number
31 is a Prime Number
37 is a Prime Number
41 is a Prime Number
43 is a Prime Number
47 is a Prime Number
53 is a Prime Number
59 is a Prime Number
61 is a Prime Number
67 is a Prime Number
71 is a Prime Number
73 is a Prime Number
79 is a Prime Number
83 is a Prime Number
89 is a Prime Number
97 is a Prime Number
```

Figure 5 Code + Output for Question 3 (Hands On)

Question 4: **Serialization**

Write a programme to serialize 3 fields out of 5 and deserialize it. Use UUID to prevent object mutation.

Solution:

```java
import java.util.*;

class Data implements Serializable {
    UUID serialversionUID = new UUID(100,10);
    transient int id;
    transient int age;
    String name;


    public Data(int id, String name, int age)
    {
        this.id = id;
        this.name = name;
        this.age = age;
    }
}
public class Question4{
    public static void display(Data d){
        System.out.println("ID Data: " + d.id);
        System.out.println("Name Data: " + d.name);
        System.out.println("Age Data: " + d.age);
    }
    public static void main(String []args){
        Data d1 = new Data(1, "Vishnu Singh Sengar", 21);
        try {
            System.out.println("\nSerializing\n");
            FileOutputStream file = new FileOutputStream("dataFile.txt");
            ObjectOutputStream out = new ObjectOutputStream(file);
            out.writeObject(d1);
            out.close();
            file.close();
            System.out.println("Data before deserialization:\n ");
            display(d1);
        }
        catch (Exception e) {
            System.out.println("Exception Caught");
        }
        d1 = null;
        try {
            System.out.println("\nDeserializing\n");
            FileInputStream file = new FileInputStream("dataFile.txt");
            ObjectInputStream in = new ObjectInputStream(file);
            d1 = (Data) in.readObject();
            in.close();
            file.close();
            System.out.println("\n\nData after deserialization:\n ");
            display(d1);
        }
        catch (Exception e) {
            System.out.println("Exception Caught");
        }
```

```
<terminated> Question4 [Java Application] C:\Users\vishn\AppData\Local\Temp\eoi46

Serializing

Data before deserialization:

ID Data: 1
Name Data: Vishnu Singh Sengar
Age Data: 21

Deserializing



Data after deserialization:

ID Data: 0
Name Data: Vishnu Singh Sengar
Age Data: 0
```

Figure 6 Code + Output for Question 4 (Hands On)

The serialized data file:

¬í sr handsOnQuestions.DataÓ;í-!Â] L namet Ljava/lang/String;L serialversionUIDt Ljava/util/UUID;xpt
 Vishnu Singh Sengarsr
java.util.UUID¼™ ÷˜m…/ J
leastSigBitsJ
mostSigBitsxp

d

End of Document