SIES Graduate School of Technology

Subject : OOPM

Roll No.: 119A1076

Assignment No. 1

Date : 11/11/2020

Q1) Program Structure of Java

Documentation : One can write comments here to understand code better.

Package statements : A package is a group of classes defined by a name. In this, you can call or declare the package that will be used in the code

Import statement : Importing certain package/classes

Class Definition : classes are the main & essential elements of any Java program

Main Method : Every Java program require a main method as starting point of program

Execution process

i) The '.java' file is passed through the compiler, which then encode the source code into bytecode

ii) The content of each class is stored in a separate class file

Execution

i) The main class is loaded into the memory by passing '.class' file to JVM. All other classes are loaded through class loads.

ii) After bytecode is loaded, it has to be ~~inserted~~ inspected by bytecode verifier who checks that the instruction don't perform damaging action

iii) Now the loaded bytecode is converted into machine code for execution

Q2) In package ABC

~~Package~~

```
package ABC
public class Operation{
    public void add(int a, int b){ int sum = a + b;
                                    SOP(sum);}
```

```
    public void sub(int a, int b){
        int sub = a-b;
        sop(sub);
    }
}
Main class
~~import~~
import ABC.*;
public class Package{
    psvm(String[] args){
        operations o= new operations();
        o.add(5,5);
        o.sub(6,3);
    }
}
```

3)

```
class complex {
    int real, imaginary;
    complex(){};

    complex(int tempReal, int tempImag)
    {
        real= tempReal;
        imaginary= tempImag;
    }
    complex(int tempReal){
        real= tempReal
        imaginary= tempReal
    }
    Complex add (complex c1, complex c2){
        complex temp = new complex();
        temp.real = c1.real + c2.real;
        temp.imaginary = c1.imaginary + c2.imaginary;
        return temp;
    }
```

```
Complex sub (Complex C1, Complex C2){
    Complex temp= new Complex();
    temp.real= C1.real - C2.real;
    temp.imaginary = C1.imaginary - C2.imaginary;
    return temp;
}
public static void main(String[] args){
    Complex C1 = new Complex(5, 7);
    Complex C2 = new Complex(2)

    Complex C3 = new Complex();
     C3 = C3.add(C1, C2);
     SOP(C3.real + "+" + C3.imaginary + "j");

    Complex C4 = new Complex();
    C4= C4.sub(C1, C2);
    SOP(C4.real + "+" + C4.imaginary + "j");
  }
}
```

Q)

```
class Count{
    psvm(String[] args){
        int upper=0, number=0;
        int lower =0, specialchar =0;

        Scanner S = new Scanner(--);
        String str;
        str = s.nextLine
        for (int i=0; i< str.length(); i++){
            char ch = str.charAt(i);
            if (ch >= 'A' && ch <='Z')
                upper ++;
            else if (ch >= 'a' && ch == 'z')
                lower ++;
```

```
        else if (ch >= '0' && ch <= '9')
            number++;
        else
            specialchar++;
    }
    SOP ("lower" + lower);
    SOP ("upper" + upper);
    SOP ("Number" + number);
}
```

Q5)

a) Wrapper class

A wrapper class is one whose object wrap or contains primitive data types. When we create an object to a wrapper class, it contains a ~~Fie~~ field & in this we can warp primitive value into wrapper class object

b) String Buffer class

A string Buffer class is similar to string but has a lot more functionality strings are fixed, immutable, but a string Buffer represent characters sequence