Dt: 9/12/2020

Execution flow of above program:

ClassFiles:

  Addition.class
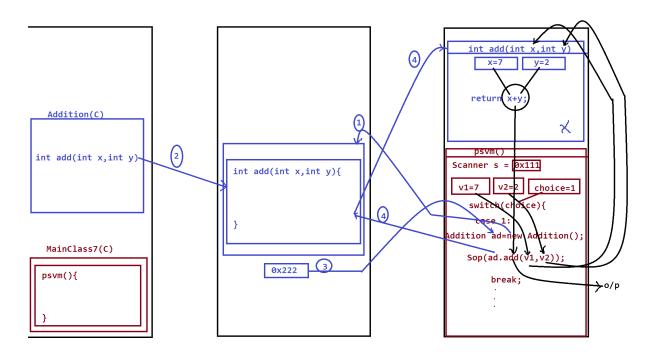
  Subtraction.class

  Multiplication.class

  Division.class

  ModDivision.class

  MainClass7.class

Note:

  =>In the above program v1,v2,x and y are known as parameters because

    they are used to transfer the data from one method to another

    method.

  =>v1 and v2 are known as Actual parameters because they hold original

input data.

=>x and y are known as Formal parameters because they are declared
part of Method signatures and holds intermediate data.

=>we can have same names for Actual parameters and Formal parameters.

=>we pass parameters to the methods while method_call.

=>In the above program the required SubClass is loaded for execution
which is needed in Object creation process.which saves the loading time
of execution process and generates HighPerformance.

Exp program:

```java
import java.lang.System;

import java.lang.String;

import java.util.Scanner;

class BranchCheck //SubClass

{
        boolean k=false;

    boolean verify(String br)

        {

                switch(br)

                {

                        case "CSE":k=true;

                        break;

                        case "EEE":k=true;

                        break;

                        case "ECE":k=true;
```

```java
                                    break;
                }//end of switch
                return k;
        }
}
class RollNoValidate //SubClass
{
        boolean z=false;
        String branch=null;
        boolean verify(String br,String code)
        {
                switch(code)
                {
                        case "05":branch="CSE";
                        break;
                        case "02":branch="EEE";
                        break;
                        case "04":branch="ECE";
                        break;
                }//end of switch
                if(branch!=null)
                {
        if(branch.equals(br))
                        {
                          z=true;
```

```java
                }
        }
        return z;
    }
}
class SResult //SubClass
{
    float per;
    String result;
    void cal(int p,int totM)
    {
        per=(float)totM/6;
        if(p==1)
        {
            result="Fail";
        }
        else if(per>=70 && per<=100)
        {
            result="Distinction";
        }
        else if(per>=60 && per<70)
        {
            result="FirstClass";
        }
        else if(per>=50 && per<60)
```

```java
                {
                        result="SecondClass";
                }
                else if(per>=35 && per<50)
                {
                        result="ThirdClass";
                }
        }
        void getResult()
        {
System.out.println("Per:"+per+"\nResult:"+result);
        }
}
class MainClass8 //MainClass
{
        public static void main(String[] args)
        {
Scanner s = new Scanner(System.in);
System.out.println("Enter the StuName:");
String name = s.nextLine();
System.out.println("Enter the Branch:");
String br = s.nextLine().toUpperCase();
BranchCheck bc = new BranchCheck();
boolean k = bc.verify(br);
        if(k)
```

```java
        {
System.out.println("Enter the RollNo:");
String rollNo = s.nextLine();
    if(rollNo.length()==10)
                {
RollNoValidate rnv = new RollNoValidate();
boolean z = rnv.verify(br,rollNo.substring(6,8));
        if(z)
                        {
System.out.println("===Enter 6 Sub marks===");
int p=0,i=1,totMarks=0;
while(i<=6)
                {
System.out.println("Enter the marks of sub"+i);
int sub = s.nextInt();
i++;
    if(sub<0 || sub>100)
                    {
        System.out.println("Invalid Sub marks...");
        i--;
        continue;
                    }
    if(sub>=0 && sub<=34)
                    {
        p=1;
```

```java
                    }
totMarks = totMarks+sub;
            }//end of loop
System.out.println("StuName:"+name);
System.out.println("Branch:"+br);
System.out.println("RollNo:"+rollNo);
System.out.println("totMarks:"+totMarks);
SResult sr = new SResult();
sr.cal(p,totMarks);
sr.getResult();
                }//end of if
                    else
                    {
System.out.println("RollNo not matched with branch...");
                    }
                }//end of if
                else
                {
System.out.println("Invalid rollNo...");
                }
            }//end of if
            else
            {
System.out.println("InValid barnch...");
            }
```

```
        }
}
```