

Batch:B4**Roll No.: 16010122221****Experiment / assignment / tutorial No. 02****Grade: AA / AB / BB / BC / CC / CD /DD****Signature of the Staff In-charge with date****TITLE : Control Statement**

AIM: Create a class myMath. The class contains the following static methods.

i) power (x, y) – to compute x^y

ii) fact (x) – to compute $x!$

Write a program to find the following series.

- $e^x = 1 + (x/1!) + (x^2/2!) + (x^3/3!) + (x^4/4!) + \dots$ upto n terms (n given by user).
- $(1+x)^n = 1 + (nx/1!) + ((n(n-1)x^2)/2!) + \dots$ upto n terms (n given by user).

(Do not make use of inbuilt functions. Use the functions of user defined class MyMath.)

Expected OUTCOME of Experiment:

CO2: Explore arrays, vectors, classes and objects in C++ and Java.

Books/ Journals/ Websites referred:

1. E. Balagurusamy , “Programming with Java” McGraw-Hill.

2. Sachin Malhotra, Saurabh Choudhary, "Programming in Java", Oxford Publications.

Pre Lab/ Prior Concepts

Java basic constructs (like if else statement, control structures, and data types Programming languages provide various control structures that allow for more complicated execution paths.

A loop statement allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages –

Sr.No.	Loop & Description
1	<u>while loop</u> Repeats a statement or group of statements while a given condition is true. It tests the condition before executing the loop body.
2	<u>for loop</u> Execute a sequence of statements multiple times and abbreviates the code that manages the loop variable.
3	<u>do...while loop</u> Like a while statement, except that it tests the condition at the end of the loop body.

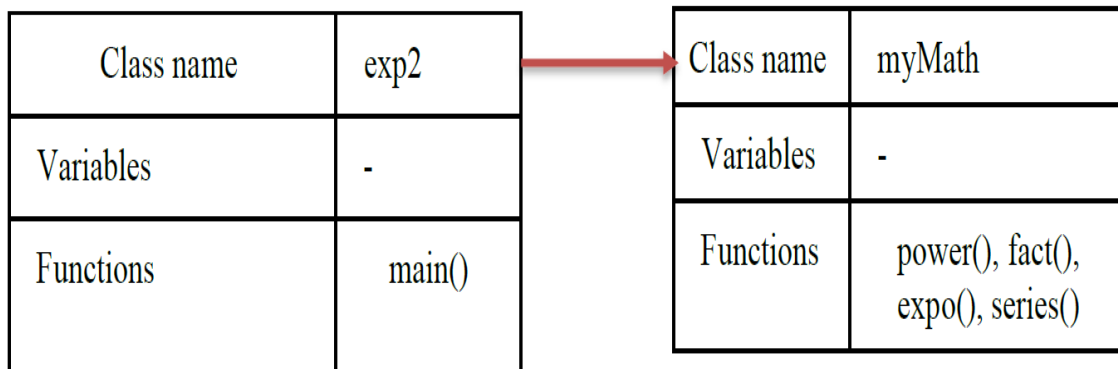
Loop Control Statements

Loop control statements change execution from its normal sequence. When execution leaves a scope, all automatic objects that were created in that scope are destroyed.

Java supports the following control statements. Click the following links to check their details.

Sr.No.	Control Statement & Description
1	<u>break statement</u> Terminates the loop or switch statement and transfers execution to the statement immediately following the loop or switch.

2	<u>continue statement</u> Causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating.
---	--

Class Diagram:**Algorithm:**

1. Start
2. Print 2^5 power : call power()
3. Print 5 factorial : call fact()
4. Take user input for n and x
5. Print result $(1+x)^n$: call series()
6. Print e^x result : call expo()

power()

1. Start
2. x and n are parameters
3. Initialize ans = 1
4. Loop n times
 - a) Ans = ans * x
5. return ans

fact()

1. Start
2. n is a parameter
3. if $n > 1$
 - return $n * \text{fact}(n-1)$
4. else return 1.

series()

1. Start
2. x and n are parameters
3. initialize ans = 0
4. loop $n+1$ times
 - ans = ans + (power(x, i) * (fact(n) / fact(n - i))) / fact(i)
5. return ans

expo()

1. Start
2. x and n are parameters
3. initialize ans = 0
4. loop n times
 - ans = ans + (power(x, i) / fact(i))
5. return ans

Implementation details:

```
import java.util.*;
import java.lang.Math;
public class Maclaurin_series {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("enter value of x for maclaurin");
        int x=sc.nextInt();
        System.out.println("enter value of n");
        int n=sc.nextInt();
        System.out.println("enter value of x for expansion series between 0
and 1 ");
        double y=sc.nextDouble();

        double e = 1, m = 1;
        for (int i = 1; i <= n; i++)
            e += MyMath.power(x, i) / MyMath.fact(i);
        System.out.println("\ne^" + x + " = " + e);
        double r2=Math.exp(x);
        System.out.println("expected e^"+x+"="+r2);
        for (int i = 1; i <= n; i++)
            m += (MyMath.power(y, i) * MyMath.fact(n)) / (MyMath.fact(i) *
MyMath.fact(n - i));
        System.out.println("\n(1+ " + y + ")^" + n + " = " + m);
        double r1=Math.pow((1+y),n);
        System.out.println("expected (1+"+y+")^"+n+"="+r1);
    }
}

class MyMath {

    public static double power(double x, int y) {
        if (y == 0)
            return 1;
        else
            return power(x, y - 1) * x;
    }

    public static double fact(int x) {
        if (x == 0)
            return 1;
        else
            return fact(x - 1) * x;
    }
}
```

```
}  
}
```

Output:

```
enter value of x for maclaurin  
2  
enter value of n  
50  
enter value of x for expansion series between 0 and 1  
0.25  
  
e^2 = 7.389056098930649  
expected e^2=7.38905609893065  
  
(1+0.25)^50 = 70064.92321624083  
expected (1+0.25)^50=70064.92321624086
```

Conclusion:

Developed custom math functions in java and implement it using static class concept.

Successfully executed the given problem statement.

Date: _____**Signature of faculty in-charge**

Post Lab Descriptive Questions

Q.1 Write a program to find the largest of three numbers using the if-else construct.

```
import java.util.Scanner;
class exp2post {
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
System.out.println("Enter 3 numbers");
double a=sc.nextDouble();
double b=sc.nextDouble();
double c=sc.nextDouble();
if(a>b){
if(a>c)
System.out.println(a+" is the largest");
else
System.out.println(c+" is the largest");
}
else{
if(b>c)
System.out.println(b+" is the largest");
else
System.out.println(c+" is the largest");
}
}
}
```

Output Clear

```
java -cp /tmp/Z8IVseJ0Mi exp2post
Enter 3 numbers
10
-5
70
70.0 is the largest
```

Q.2 Write a program to determine the sum of the following series for a given value of n:

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

```
import java.util.Scanner;
class ex2post2 {
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
System.out.print("Enter n: ");
int n=sc.nextInt();
double ans=0;
for(int i=1;i<=n;i++)
ans=ans + 1.0/i;
System.out.println("The answer is: "+ans);
}
}
```

Output**Clear**

```
java -cp /tmp/Z8IVseJ0Mi ex2post2
Enter n: 5
The answer is: 2.283333333333333
```