Name:KUSUM M R **Date:24/11/2020**
USN: 1BM19CS077

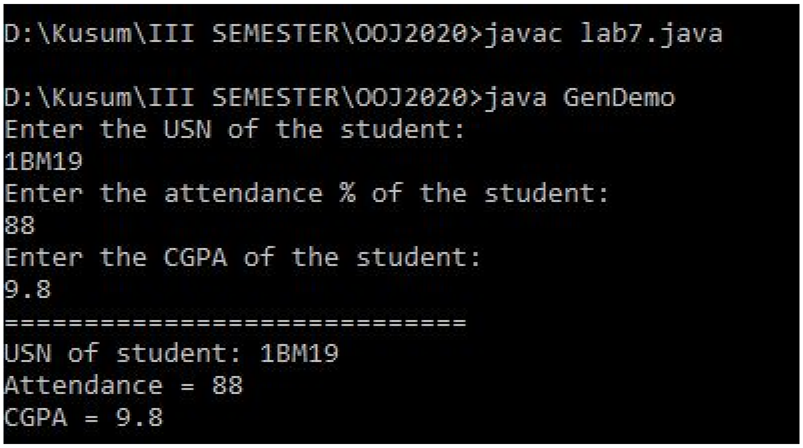
LAB 7:

Write a program to demonstrate generics with multiple object parameters.

SOURCE CODE:

```
import java.util.*;
class Gener<T,U,V>
{
	T usn;
	U attendance;
	V cgpa;
	Gener(T n,U a,V c)
	{
		usn = n;
		attendance = a;
		cgpa = c;
	}
	void display()
	{
		System.out.println("=============================");
		System.out.println("USN of student: "+usn);
		System.out.println("Attendance = "+attendance);
		System.out.println("CGPA = "+cgpa);
	}
}
class GenDemo
{
	public static void main(String args[])
	{
		Scanner in = new Scanner(System.in);
		String USN;
		int attd;
		double cg;
		System.out.println("Enter the USN of the student:");
		USN = in.next();
		System.out.println("Enter the attendance % of the student:");
		attd = in.nextInt();
		System.out.println("Enter the CGPA of the student:");
		cg = in.nextDouble();
		Gener<String, Integer, Double>  ob = new Gener<String, Integer, Double>(USN, attd, cg);
		ob.display();
	}
}
```

OUTPUT:



```
D:\Kusum\III SEMESTER\OOJ2020>javac lab7.java

D:\Kusum\III SEMESTER\OOJ2020>java GenDemo
Enter the USN of the student:
1BM19
Enter the attendance % of the student:
88
Enter the CGPA of the student:
9.8
=============================
USN of student: 1BM19
Attendance = 88
CGPA = 9.8
```

LAB 8:

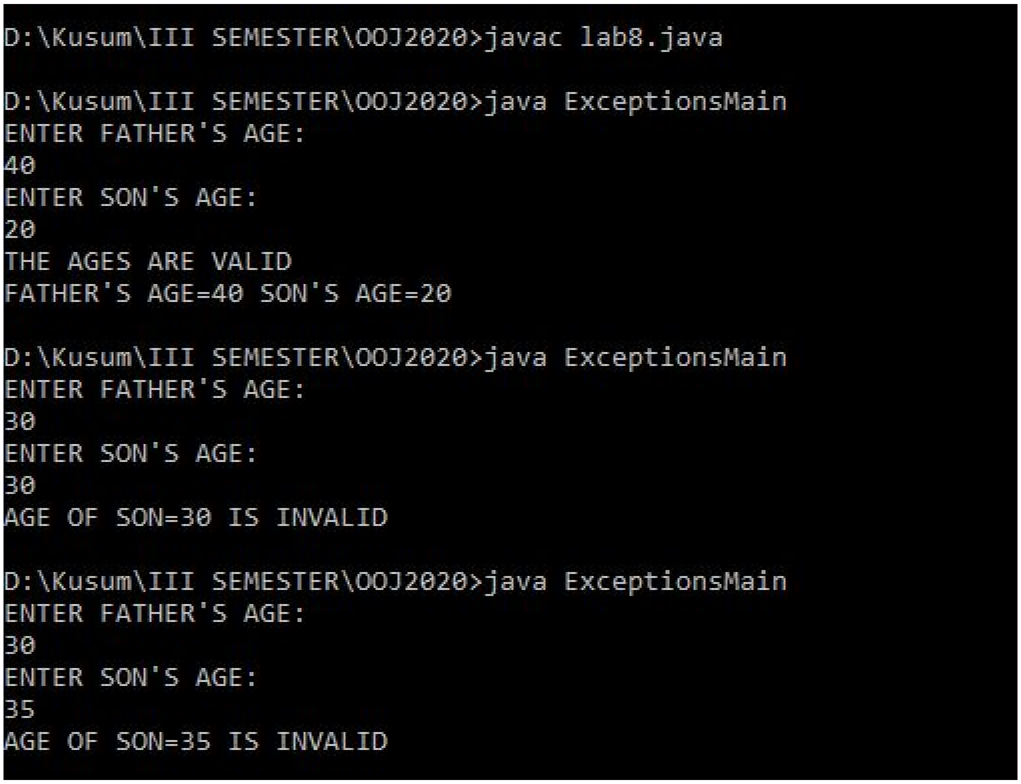
8. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called Father and derived class called Son which extends the base class. In Father class, implement a constructor which takes the age and throws the exception Wrong Age( ) when the input age=father's age.

SOURCE CODE:

```
import java.util.Scanner;

class WrongAge extends Exception {
    int age;

    WrongAge(int x) {
        age = x;
    }

    public String toString() {
        return "AGE OF SON=" + age + " IS INVALID";
    }
}

class Father {
    int a;

    Father(int x) {
        a = x;
    }
}
```

```
class Son extends Father {
    int age;

    Son(int fage, int sage) {
        super(fage);
        age = sage;
    }

    void compute() throws WrongAge {
        if (age >= a) {
            throw new WrongAge(age);
        } else {
            System.out.println("THE AGES ARE VALID");
            System.out.println("FATHER'S AGE=" + a + "\t" + "SON'S AGE=" + age);
        }
    }

}

class ExceptionsMain {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        System.out.println("ENTER FATHER'S AGE:");
        int f = s.nextInt();
        System.out.println("ENTER SON'S AGE:");
        int so = s.nextInt();
        Son ss = new Son(f, so);
        try {
            ss.compute();
        } catch (WrongAge e) {
            System.out.println(e);
        }
    }
}
```

OUTPUT:



```
D:\Kusum\III SEMESTER\OOJ2020>javac lab8.java

D:\Kusum\III SEMESTER\OOJ2020>java ExceptionsMain
ENTER FATHER'S AGE:
40
ENTER SON'S AGE:
20
THE AGES ARE VALID
FATHER'S AGE=40 SON'S AGE=20

D:\Kusum\III SEMESTER\OOJ2020>java ExceptionsMain
ENTER FATHER'S AGE:
30
ENTER SON'S AGE:
30
AGE OF SON=30 IS INVALID

D:\Kusum\III SEMESTER\OOJ2020>java ExceptionsMain
ENTER FATHER'S AGE:
30
ENTER SON'S AGE:
35
AGE OF SON=35 IS INVALID
```