SCHOOL OF
COMPUTING

Susendran M

CH.SC.U4CSE24154

OBJECT ORIENTED PROGRAMMING (23CSE111)

LAB RECORD

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING, CHENNAI

BONAFIDE CERTIFICATE

This is to certify that the Lab Record work for 23CSE111- Object Oriented Programming Subject submitted by ***CH.SC.U4CSE24154 – Susendran M*** in **"Computer Science and Engineering"** is a Bonafide record of the work carried out under my guidance and supervision at Amrita School of Computing, Chennai.

This Lab examination held on

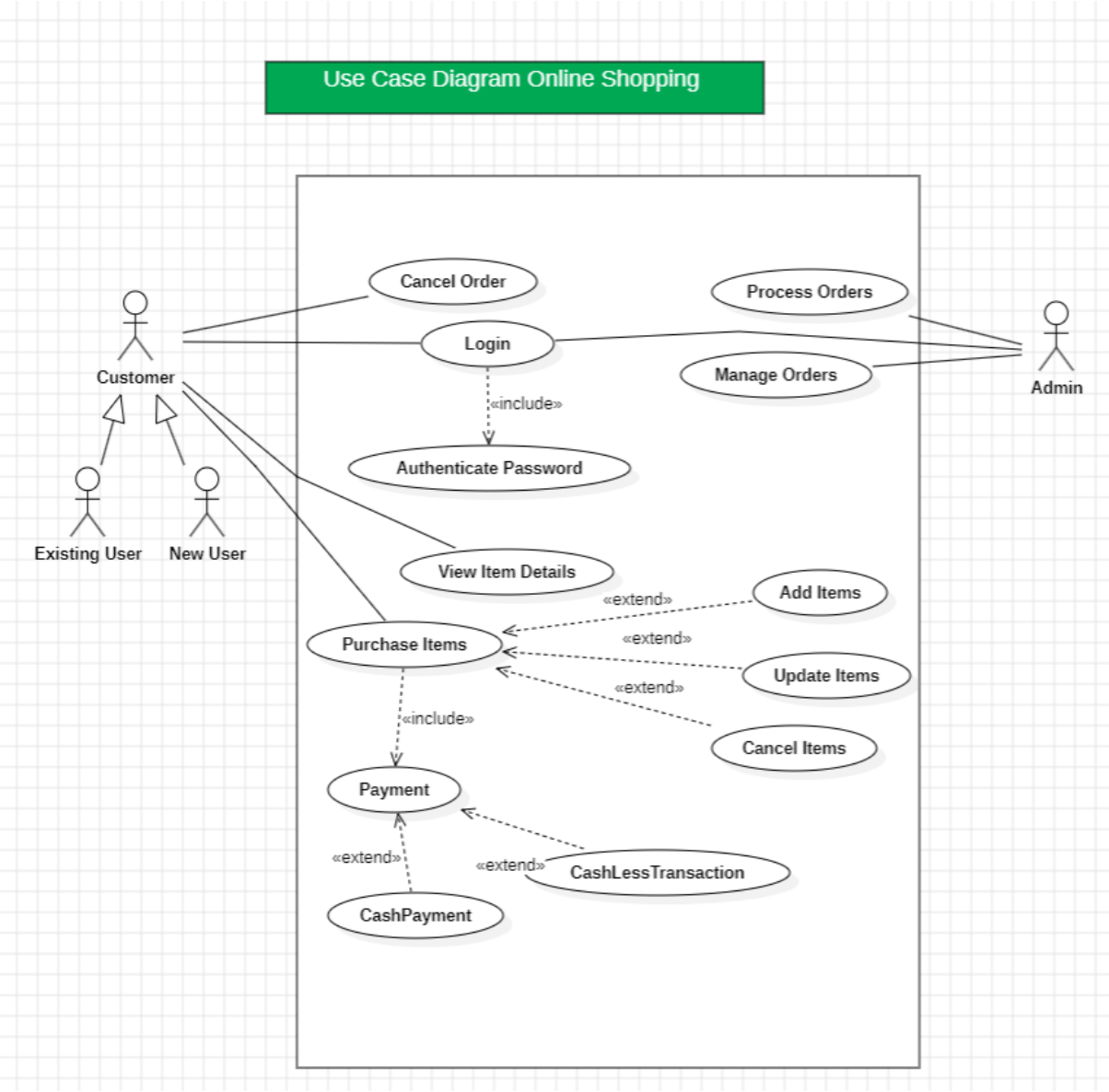
Internal Examiner 1 Internal Examiner 2

INDEX

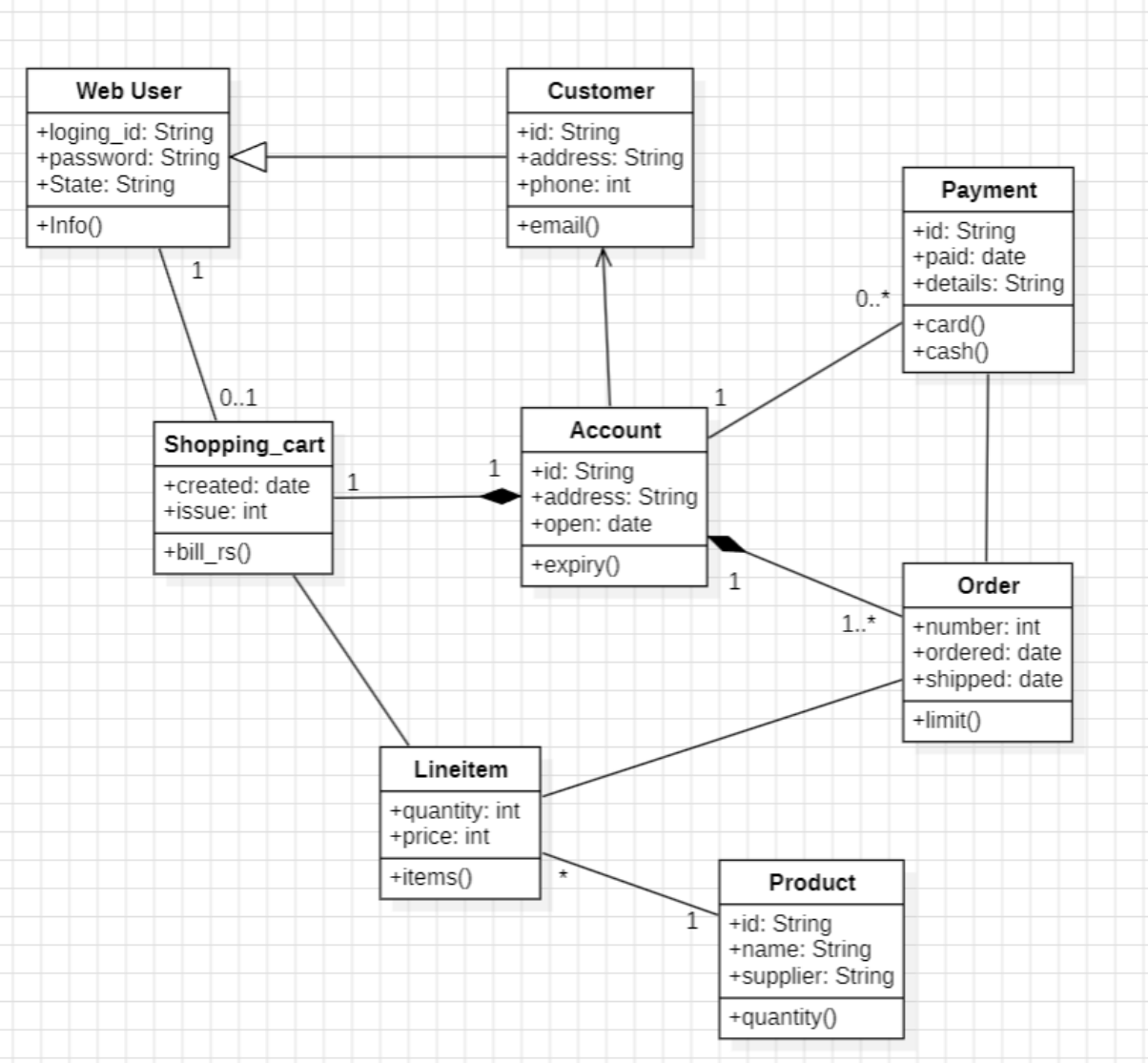
UML DIAGRAMS

1. ONLINE SHOPPING

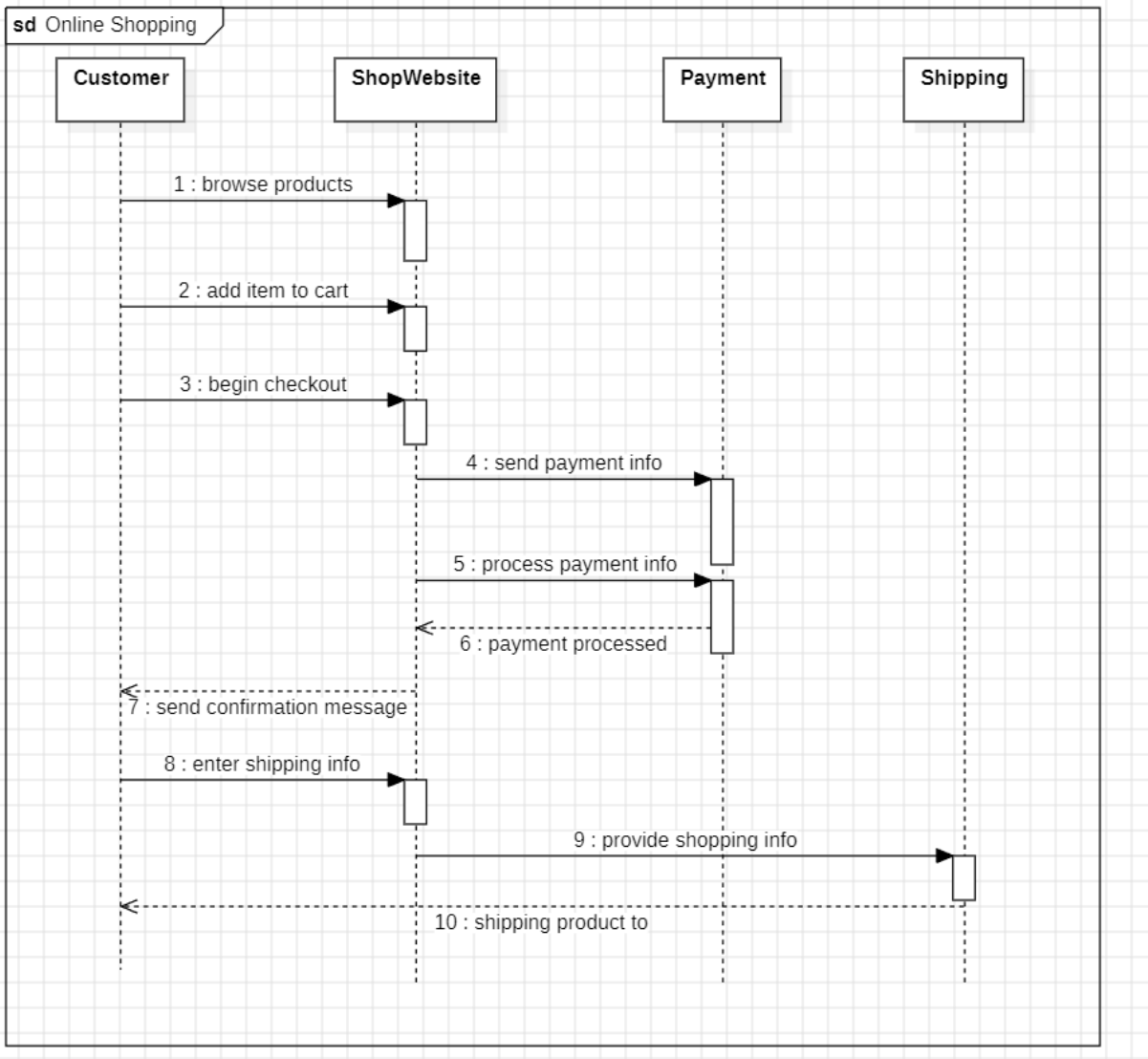
1.a) Use Case Diagram:



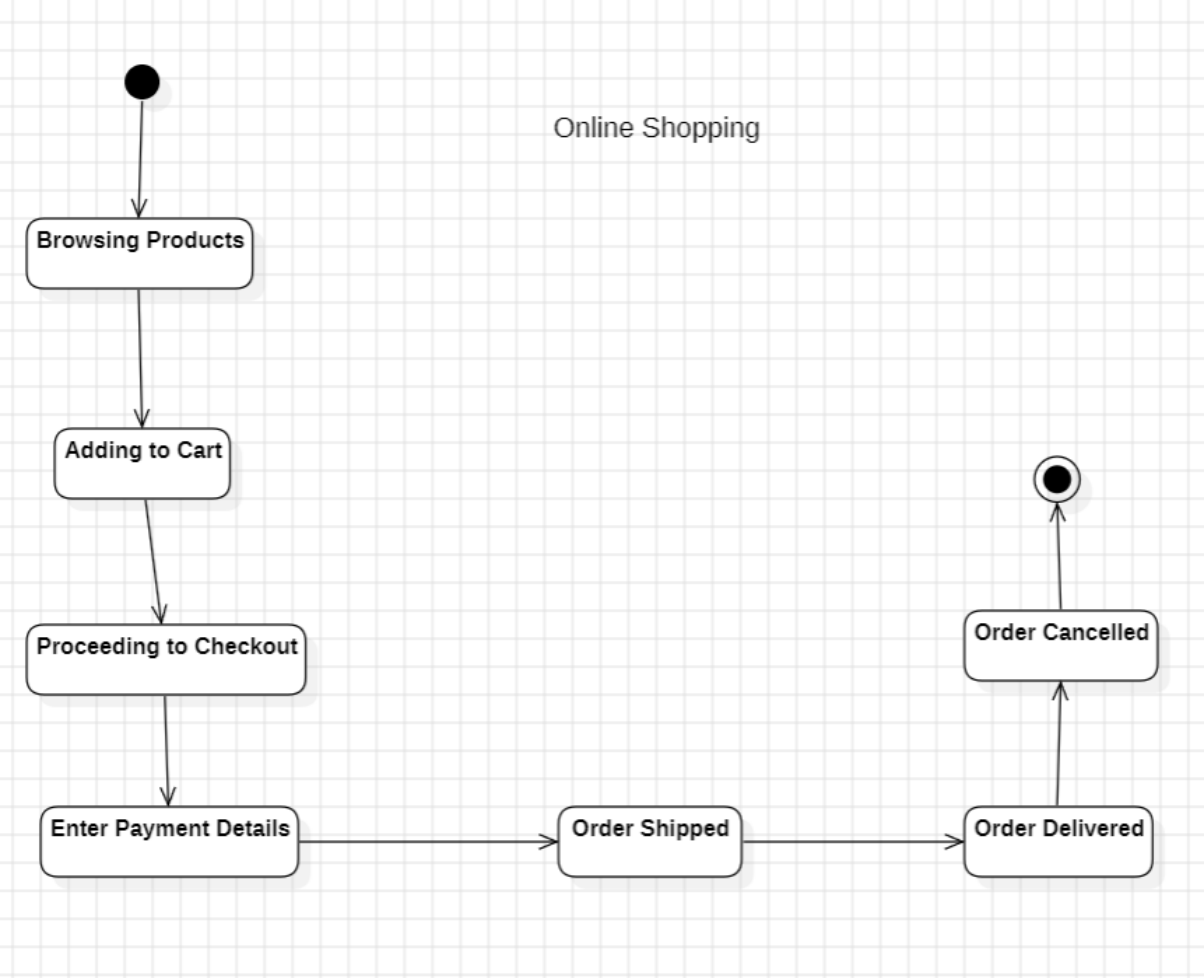
1.b) Class Diagram:



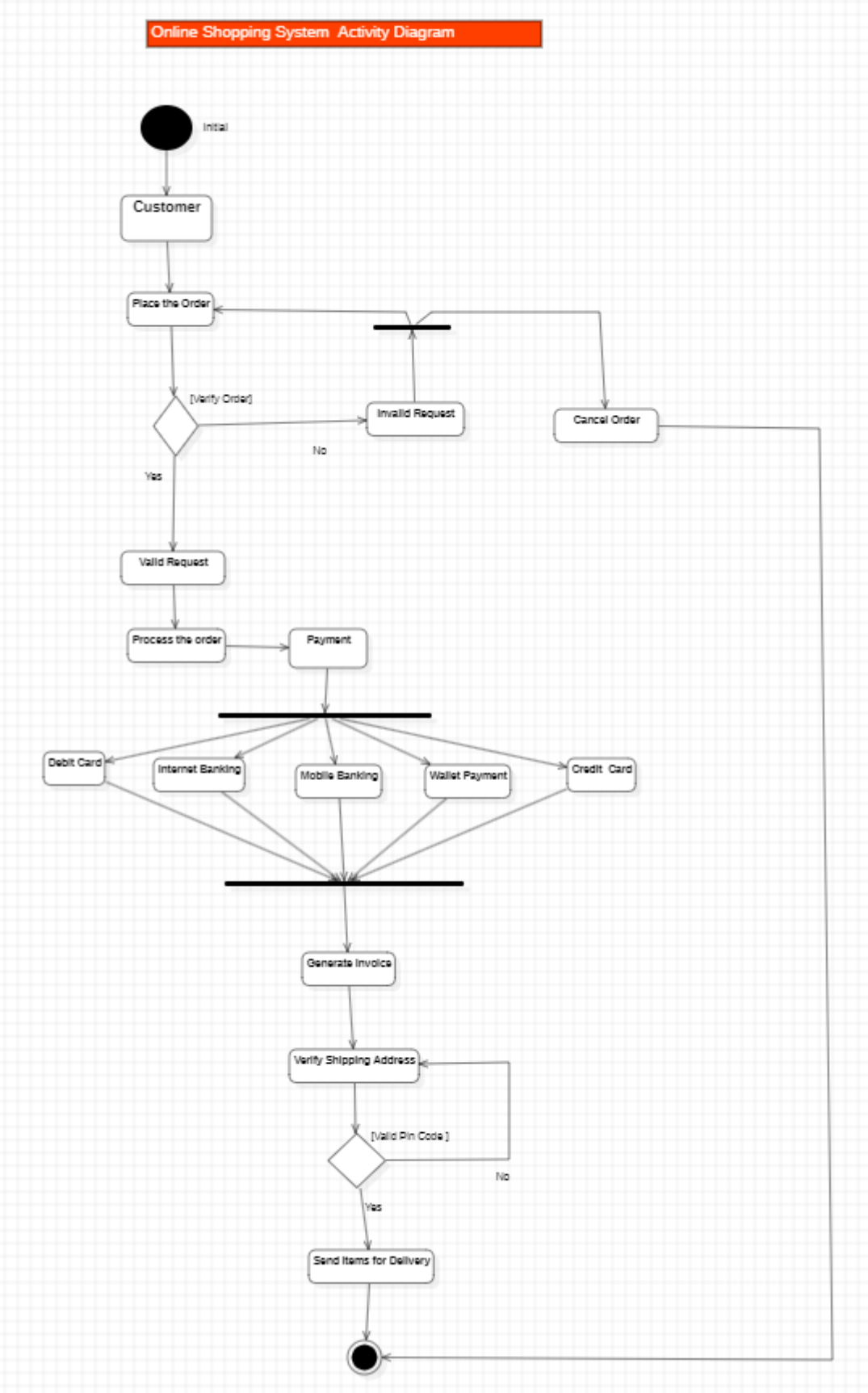
1.c) Sequence Diagram:



1.d) State Diagram:

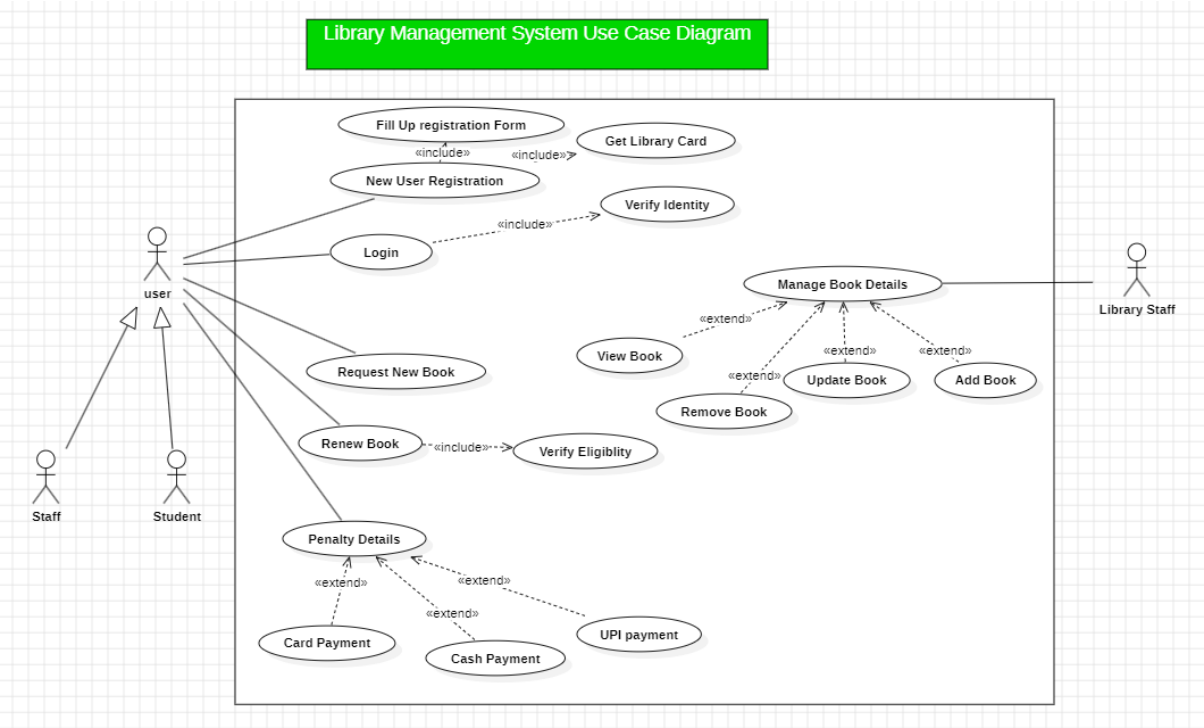


1.e) Activity Diagram:

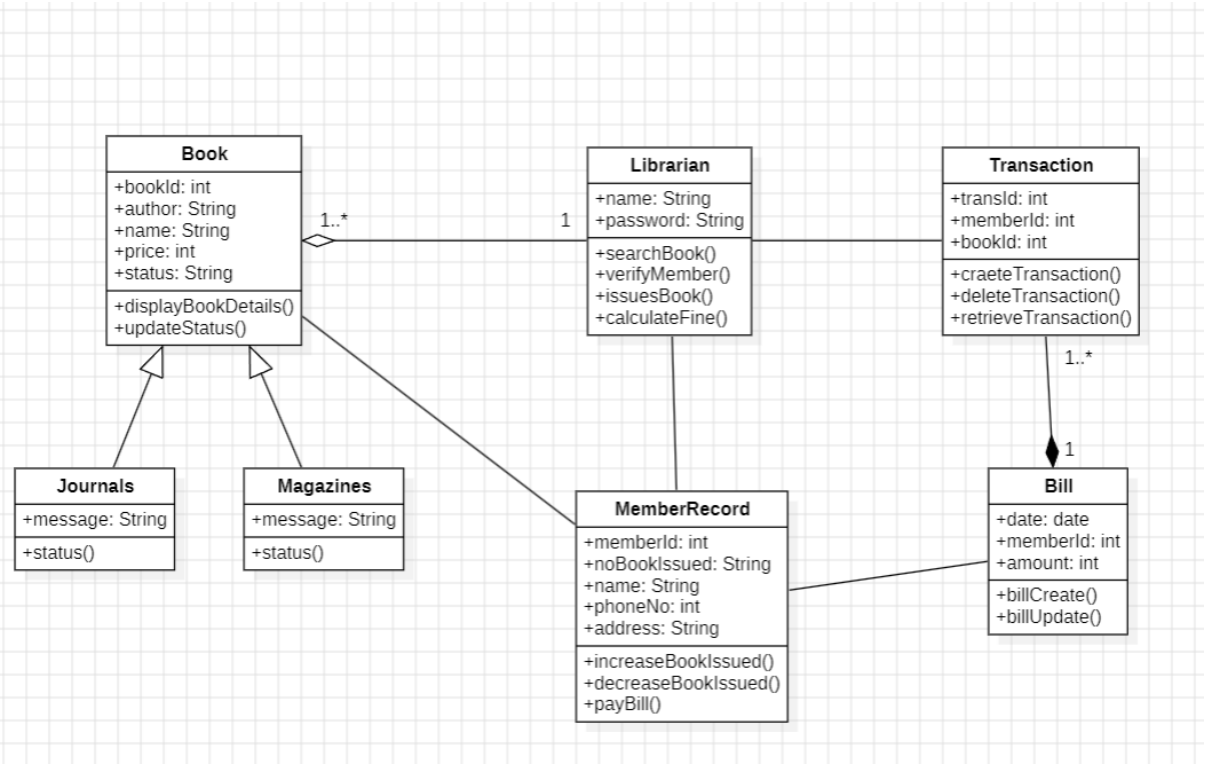


2. LIBRARY MANAGEMENT SYSTEM

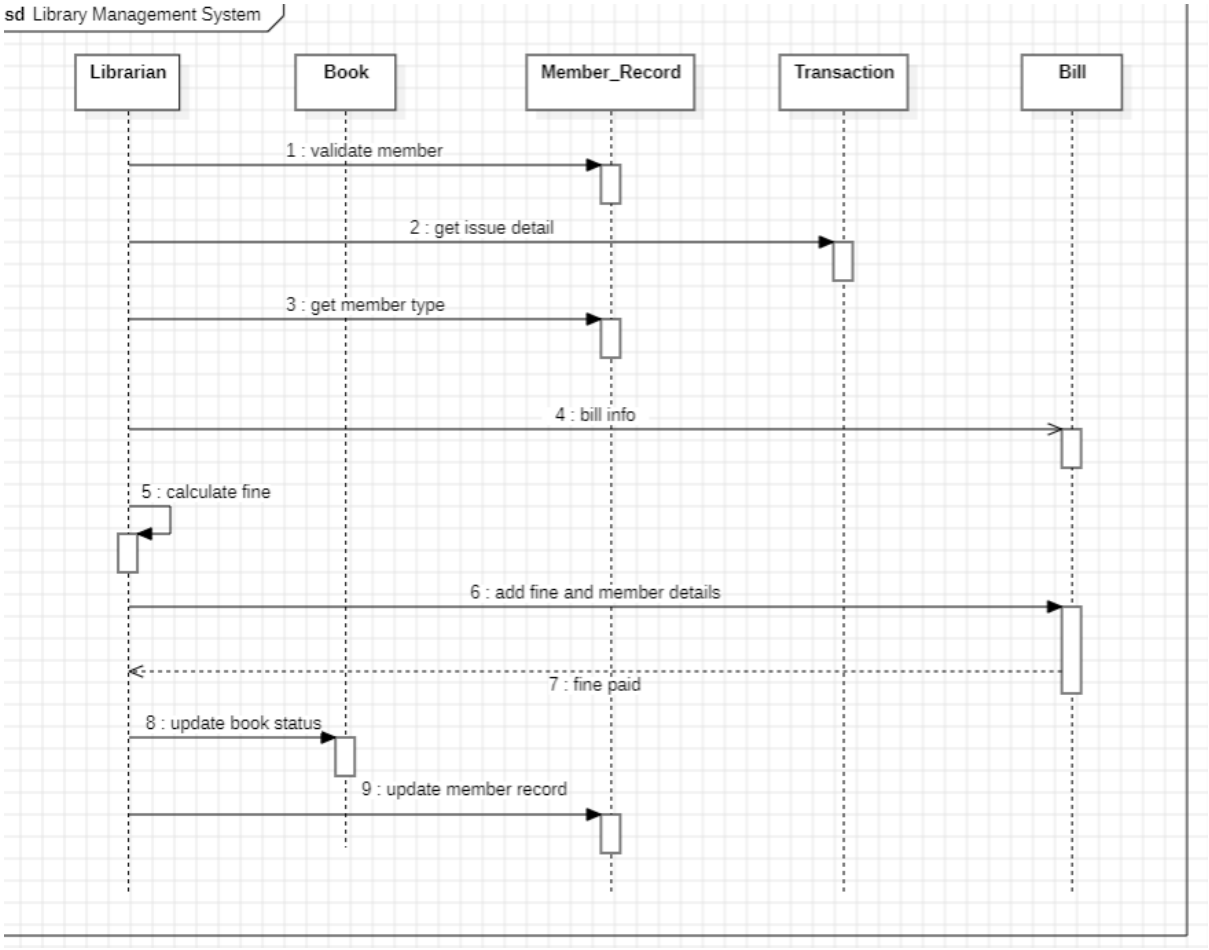
2.a) Use Case Diagram:



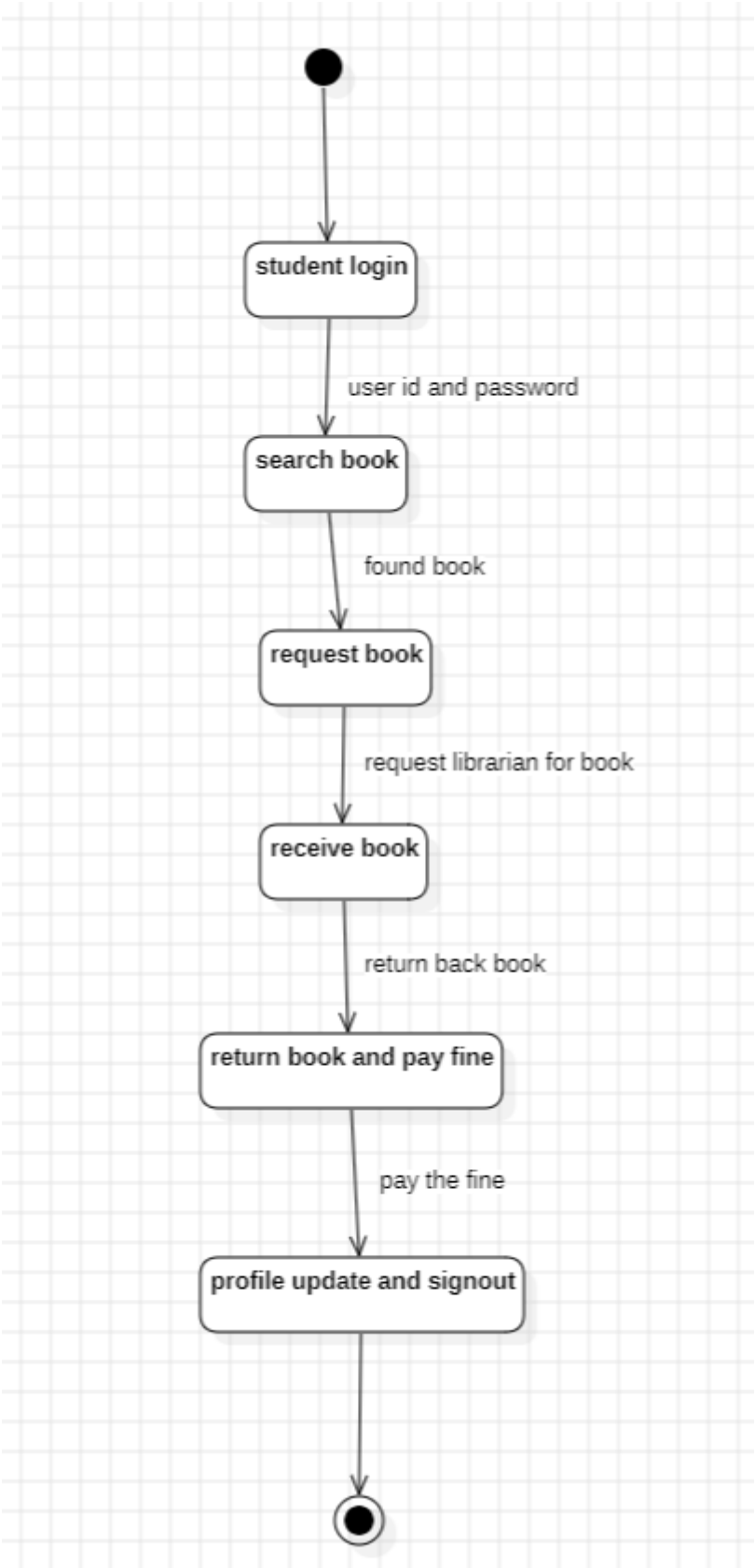
2.b) Class Diagram:



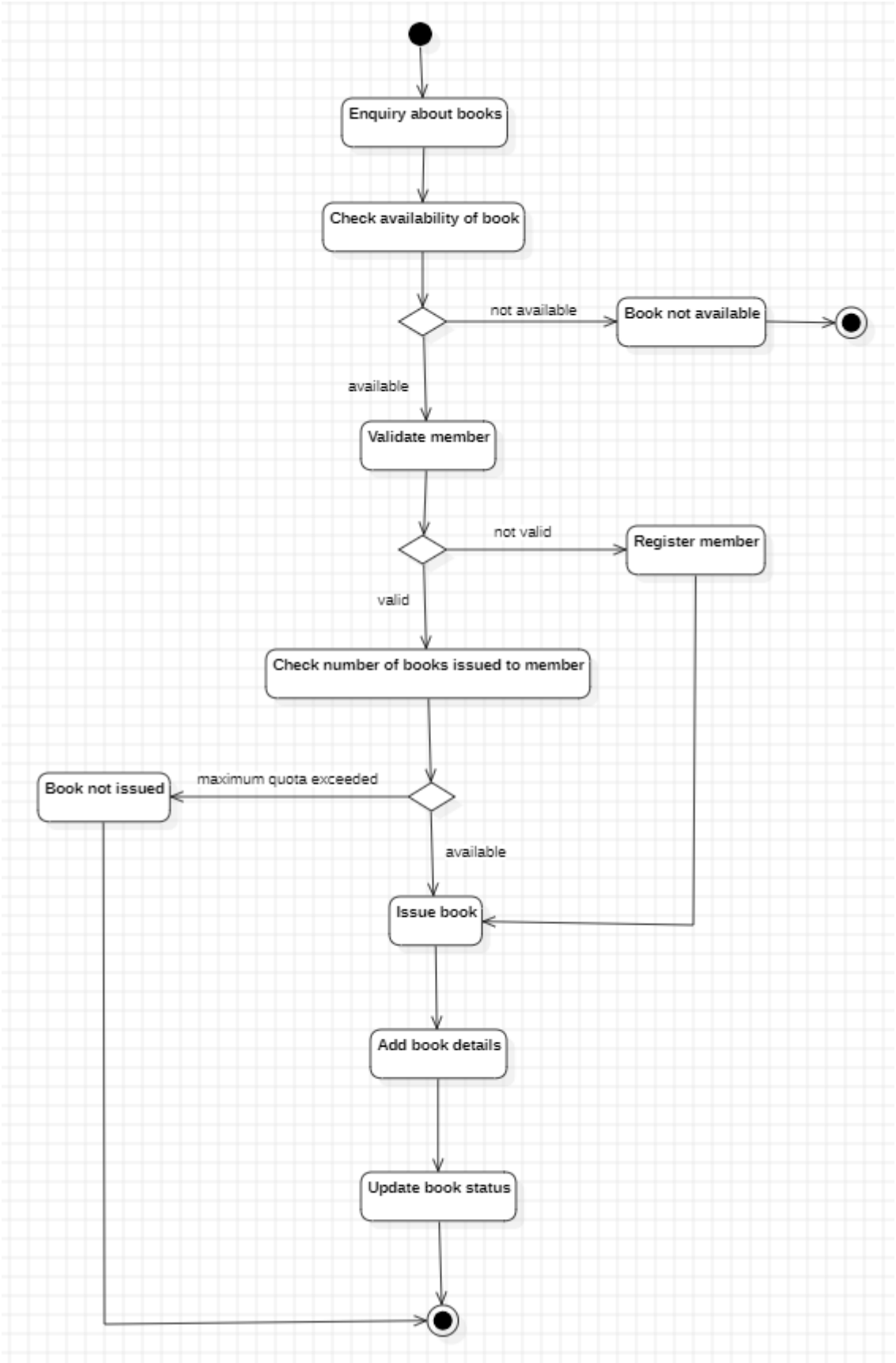
2.c) Sequence Diagram:



2.d) State Diagram:



2.e) Activity Diagram:



3. Basic Java Programs

3.a) Sum of Digits :

Code:

```
import java.util.Scanner;
public class sum_of_digits
{
int num,sum=0;
public sum_of_digits(int num)
{
this.num=num;
}
public void operation()
{
while (num != 0)
{
sum=sum+num%10;
num=num/10;
}
System.out.println("Sum of digits : " + sum);
}
public static void main (String[] args)
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
sum_of_digits obj=new sum_of_digits(inputNum);
obj.operation();
```

```
}
}
```

Output:

```
C:\Problems>javac sum_of_digits.java

C:\Problems>java sum_of_digits.java
Enter the number :
30
Sum of digits : 3
```

3.b) Reverse Number :

Code:

```
import java.util.Scanner;
public class reverse_number
{
int num,rev=0,digit;
public reverse_number(int num)
{
this.num=num;
}
public void operation()
{
while (num != 0)
{
digit=num%10;
rev=rev*10+digit;
num=num/10;
}
```

```
System.out.println("Reversed number : " + rev);
}
public static void main (String[] args)
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
reverse_number obj=new reverse_number(inputNum);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac reverse_number.java

C:\Problems>java reverse_number.java
Enter the number :
12345
Reversed number : 54321
```

3.c) Prime Number :

Code:

```
import java.util.Scanner;
public class prime_number
{
int num;
Boolean isPrime=true;
public prime_number(int num)
```

```
{
this.num=num;
}
public void operation()
{
if (num<=1)
{
isPrime=false;
}
else
{
for (int i=2;i*i<=num;i++)
{
if (num % i == 0)
{
isPrime=false;
break;
}
}
}
if (isPrime)
{
System.out.println(num + " is a prime number");
}
else
{
System.out.println(num + " is not a prime number");
}
}
public static void main (String[] args)
```

```
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
prime_number obj=new prime_number(inputNum);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac prime_number.java

C:\Problems>java prime_number.java
Enter the number :
23
23 is a prime number
```

3.d) Palindrome Number :

Code:

```
import java.util.Scanner;
public class armstrong_number
{
int num;
public armstrong_number(int num)
{
this.num = num;
}
public void operation()
```

```
{
int originalNum = num;
int sum = 0;
int digits = String.valueOf(num).length();
while (num != 0)
{
int digit = num % 10;
sum += Math.pow(digit, digits);
num /= 10;
}
if (sum == originalNum)
{
System.out.println(originalNum + " is an Armstrong number.");
}
else
{
System.out.println(originalNum + " is not an Armstrong number.");
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
armstrong_number obj = new armstrong_number(inputNum);
obj.operation();
}
}
```

Output;

```
C:\Problems>javac palindrome_number.java

C:\Problems>java palindrome_number.java
Enter the number:
123
123 is not a palindrome number.
```

3.e) Lower Triangle :

Code:

```
import java.util.Scanner;
public class lower_triangle
{
int rows;
public lower_triangle(int rows)
{
this.rows = rows;
}
public void operation()
{
for (int i = 1; i <= rows; i++)
{
for (int j = 1; j <= i; j++)
{
System.out.print(j + " ");
}
System.out.println();
}
```

```
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number of rows: ");
int inputRows = inp.nextInt();
lower_triangle obj = new lower_triangle(inputRows);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac lower_triangle.java

C:\Problems>java lower_triangle.java
Enter the number of rows:
3
1
1 2
1 2 3
```

3.f) LCM Numbers :

Code:

```
import java.util.Scanner;
public class lcm_numbers
{
int num1, num2;
public lcm_numbers(int num1, int num2)
{
```

```
this.num1 = num1;
this.num2 = num2;
}
public void operation()
{
int lcm = (num1 > num2) ? num1 : num2;
while (true)
{
if (lcm % num1 == 0 && lcm % num2 == 0)
{
System.out.println("LCM of " + num1 + " and " + num2 + " is: " +
lcm);
break;
}
lcm++;
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the first number: ");
int inputNum1 = inp.nextInt();
System.out.println("Enter the second number: ");
int inputNum2 = inp.nextInt();
lcm_numbers obj = new lcm_numbers(inputNum1, inputNum2);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac lcm_numbers.java

C:\Problems>java lcm_numbers.java
Enter the first number:
3
Enter the second number:
4
LCM of 3 and 4 is: 12
```

3.g) Fibonacci Series :

Code:

```
import java.util.Scanner;
public class armstrong_number
{
int num;
public armstrong_number(int num)
{
this.num = num;
}
public void operation()
{
int originalNum = num;
int sum = 0;
int digits = String.valueOf(num).length();
while (num != 0)
{
int digit = num % 10;
sum += Math.pow(digit, digits);
```

```
num /= 10;
}
if (sum == originalNum)
{
System.out.println(originalNum + " is an Armstrong number.");
}
else
{
System.out.println(originalNum + " is not an Armstrong number.");
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
armstrong_number obj = new armstrong_number(inputNum);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac fibonacci_series.java

C:\Problems>java fibonacci_series.java
Enter the number of terms for the Fibonacci series:
4
Fibonacci Series up to 4 terms:
0 1 1 2
```

3.h) Factorial Number :

Code:

```
import java.util.Scanner;
public class armstrong_number
{
int num;
public armstrong_number(int num)
{
this.num = num;
}
public void operation()
{
int originalNum = num;
int sum = 0;
int digits = String.valueOf(num).length();
while (num != 0)
{
int digit = num % 10;
sum += Math.pow(digit, digits);
num /= 10;
}
if (sum == originalNum)
{
System.out.println(originalNum + " is an Armstrong number.");
}
else
{
System.out.println(originalNum + " is not an Armstrong number.");
}
```

```
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
armstrong_number obj = new armstrong_number(inputNum);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac factorial_number.java

C:\Problems>java factorial_number.java
Enter the number:
3
Factorial of 3 is: 6
```

3.i) Sum of Even, Odd Digits :

Code:

```
import java.util.Scanner;
public class even_odd_sum
{
int limit;
public even_odd_sum(int limit)
{
this.limit = limit;
```

```
}
public void operation()
{
int evenSum = 0, oddSum = 0;
for (int i = 1; i <= limit; i++)
{
if (i % 2 == 0)
{
evenSum += i;
}
else
{
oddSum += i;
}
}
System.out.println("Sum of even numbers up to " + limit + " is: " +
evenSum);
System.out.println("Sum of odd numbers up to " + limit + " is: " +
oddSum);
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the limit: ");
int inputLimit = inp.nextInt();
even_odd_sum obj = new even_odd_sum(inputLimit);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac even_odd_sum.java

C:\Problems>java even_odd_sum.java
Enter the limit:
5
Sum of even numbers up to 5 is: 6
Sum of odd numbers up to 5 is: 9
```

3.j) Armstrong Number :

Code:

```
import java.util.Scanner;
public class armstrong_number
{
int num;
public armstrong_number(int num)
{
this.num = num;
}
public void operation()
{
int originalNum = num;
int sum = 0;
int digits = String.valueOf(num).length();
while (num != 0)
{
int digit = num % 10;
sum += Math.pow(digit, digits);
```

```
num /= 10;
}
if (sum == originalNum)
{
System.out.println(originalNum + " is an Armstrong number.");
}
else
{
System.out.println(originalNum + " is not an Armstrong number.");
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
armstrong_number obj = new armstrong_number(inputNum);
obj.operation();
}
}
```

Output:

```
C:\Problems>javac armstrong_number.java

C:\Problems>java armstrong_number.java
Enter the number:
2345
2345 is not an Armstrong number.
```