**Session 13: SCALA - SESSION II**

Assignment 13.2

Write a Scala application to find the nth digit in the sequence.

* Write the function using standard for loop
* Write the function using recursion

**Solution**

# The Fibonacci sequence is the series of numbers,

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

The next number is found by adding up the two numbers before it.

The 2 is found by adding the two numbers before it (1+1)

The 3 is found by adding the two numbers before it (1+2),

And the 5 is (2+3),

And so on!

Example: the next number in the sequence above is 21+34 = 55

Formula,

**xn = xn-1 + xn-2**

Example,

The 8th term is the 7th term plus the 6th term: X8 = X7+X6

From the above table,

The 8th term is 21, hence the 7th term 21+the 6th term 13 = 34.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n = | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| xn = | 0 | 1 | 2 | 3 | 5 | 8 | 13 | 21 | 34 | 55 | 89 | 144 | 233 | 377 | 610 | 987 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# 1: write function using standard for loop

## Scala code

**package** Assignment13\_2

**object** fibseries

{

**def** main(args: Array[String]): Unit ={

*println*(**"Enter a number: "**)

**var** num:Int = scala.io.StdIn.readLine().toInt

**var** n1=0

**var** n2=1

**var** a: Int=0;

**var** b: Int=0;

*println*(**"Standard For loop"**)

**for**(a <-1 to num){

**val** sumOfPrevTwo = n1+n2

n1=n2

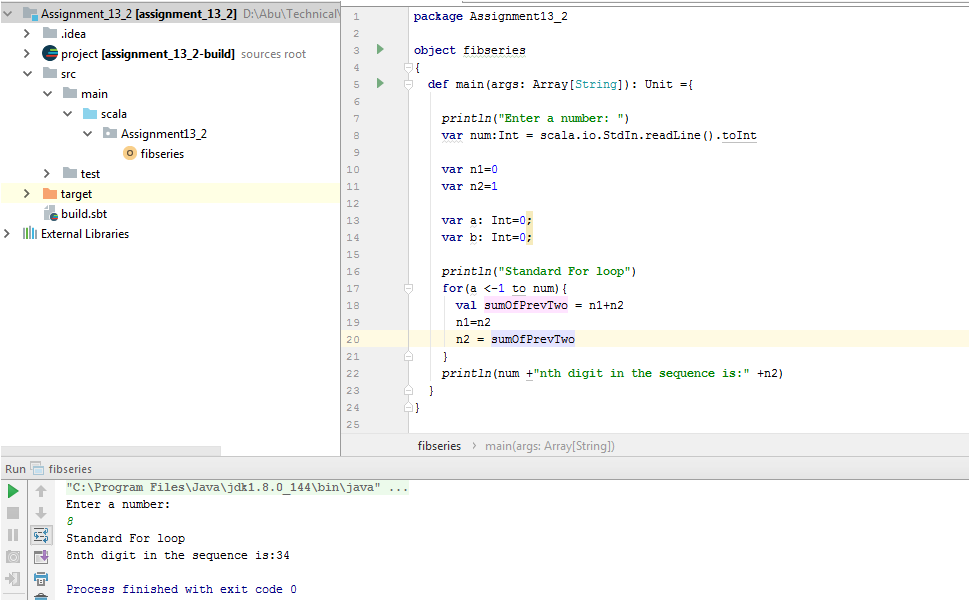
n2 = sumOfPrevTwo

}

*println*(num +**"nth digit in the sequence is:"** +n2)

}

}



# 2 - Write the function using recursion

## Scala code

**object** fibseriesrecursion

{

**def** main(args: Array[String]): Unit ={

*println*(**"Enter a number: "**)

**var** num:Int = scala.io.StdIn.readLine().toInt

*println*(**"Using Recursion"**)

*println*(num + **"nth digit in the sequence is: "** +fib(num))

**def** fib(n:Int): Int =

**if** (n<2)

1

**else**

fib(n-1+fib(n-2))

}

}

