ACD\_BDD2.3\_Session\_13\_Assignment\_2

A Fibonacci series (starting from 1) written in order without any spaces in between, thus producing a sequence of digits.

**def** fib\_seq(n: Int):*Seq*[Int] = {

**var** num1 = 0

**var** num2 = 1

**for**( i <- 1 to n) **yield** {

**var** num3 = num1 + num2

num1 = num2

num2 = num3

num1

}

}

println(fib\_seq(10))

Output

Vector(1, 1, 2, 3, 5, 8, 13, 21, 34, 55)

fib\_seq: (n: Int)Seq[Int]

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

○ Write the function using standard for loop

**val** mySeq = Seq(1,2,3,4,5,6,7,8,9,10)

**for** (i <- 0 to mySeq.length - 1)

{

println(mySeq(i))

}

output

1

2

3

4

5

6

7

8

9

10

mySeq: Seq[Int] = List(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

Write the function using recursion

**def** nthRecursive[A](n: Int, ls: *Seq*[A]): A = (n, ls) **match** {

**case** (0, h :: \_ ) => h

**case** (n, \_ :: tail) => nthRecursive(n - 1, tail)

**case** (\_, **Nil** ) => **throw** **new** *NoSuchElementException*

}

nthRecursive(6,mySeq)

Output

nthRecursive: [A](n: Int, ls: Seq[A])A

res33: Int = 5