Vijay's Assignment – Scala2

Task 1

Create a Scala application to find the GCD of two numbers

Explanation:

- GCD of two numbers is obtained by finding the largest number that will exactly divide the two numbers with zero reminder
- Two numbers considered here are 55 and 121
- The smallest of the two numbers is identified to loop till that number from 1
- From 1 till the smallest number of the two (55) the largest number that divides 55 and 121 with zero reminder is identified
- The result is displayed as shown below

```
object sct1 {
  def main(args: Array[String]): Unit ={
  var n1 = 55
  var n2 = 121
  var gcd = 0
  var div1 = 0
  var i = 0

  if (n1 < n2)
      div1 = n1
  else
      div1 = n2
  for (i<-1 to div1)
  {
    if (n1 % i == 0 && n2 %i == 0)
    {
      gcd = i
    }
  }
  println("GCD of n1:"+n1+" and n2:" +n2+ " is :"+gcd)
 }
}
```

Output

```
<terminated> sct1$ [Scala Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (17-Jan-2019, 5:03:12 PM)
GCD of n1:55 and n2:121 is :11
```

Task 2

Fibonacci series (starting from 1) written in order without any spaces in between, thus

producing a sequence of digits.

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

Writeup:

1) This logic was written to identify the N number in Fibonacci series
2) N digit will be given as input
3) Based on the task description provided, the assumption is that this code does not produce Fibonacci series but just identifies the digit at a given position in that series
4) The code starts with the first two numbers 1 & 2 and loops till 20th digit (value N) by adding the previous two digits and then progressing the two input variables to next position to proceed in the series

Write the function using standard for loop

This sample code identifies the 20th number in the Fibinocci series

```
package acad_scala1

object sct1 {
  def main(args: Array[String]): Unit ={
    /* fib example: 1235813213455891442333776109871597258441816765*/
    var a = 1
    var b = 2
    var c = 0
    var i = 0
    var n = 20
    println (n+"th digit in fibinocci series is: "+fibi())

    def fibi(): Int = {
      for (i<-3 to n) {
      c = b + a
      a = b
      b = c
     }
      c
    }

  }
}
```

Output

```
<terminated> sct1$ [Scala Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (17-Jan-2019, 5:59:23 PM)
20th digit in fibinocci series is: 10946
```

This solution identifies the same N digit using Recursion. The function “fibi” calls itself several times until the 12th digit is identified

➢ Write the function using recursion

```
def main(args: Array[String]): Unit ={
  /* fib example: 1235813213455891442333776109871597258441816765*/
  var a = 1
  var b = 2
  var c = 0
  var i = 1
  var n = 12
  println (n+"th digit in fibinocci series is: "+fibi(n))

  def fibi(n:Int): Int = {
    i = i + 1
    if (i==n)
      c
    else
    {
      c = b + a
      a = b
      b = c
      fibi(n)
   }
  }

}
```

```
<terminated> sct1$ [Scala Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (17-Jan-2019, 6:23:17 PM)
12th digit in fibinocci series is: 233
```

Task 3

Find square root of number using Babylonian method.

1. Start with an arbitrary positive start value x (the closer to the root, the better).

2.Initialize y = 1.

3. Do following until desired approximation is achieved.

a) Get the next approximation for root using average of x and y

b) Set y = n/x

Explanation

- A custom function sqrt is written to identify the square root of the input number 250
- It starts by identifying the half of the input number as the square root cannot be more than that half
- It loops from 1 to the half applying the average and division logic given in the Babylonial method until the difference between the two is 0.00001

```
package acad_scala1

object sct1 {
  def main(args: Array[String]): Unit ={
    val x = 250
    println("Square root of "+x+" is:"+sqrt(x))

    def sqrt(x:Double)={
      var n:Double = x/2
      var y:Double  =1
      val e:Double = 0.000001
      while (n-y > e)
      {
        n = (n+y)/2
        y = x/n

      }
      n
    }
   }
}
```

Output:

```
<terminated> sct1$ [Scala Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (20-Jan-2019, 3:04:44 PM)
Square root of 250 is:15.811388300842072
```