A 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.

○ Write the function using standard for loop

*scala> def fib1( n : Int) : Int = n match {*

*| case 0 | 1 => n*

*| case \_ => fib1( n-1 ) + fib1( n-2 )*

*| }*

*fib1: (n: Int)Int*

*scala> fib1(5)*

*res4: Int = 5*

*scala> fib1(10)*

*res5: Int = 55*

*scala> fib1(12)*

*res6: Int = 144*

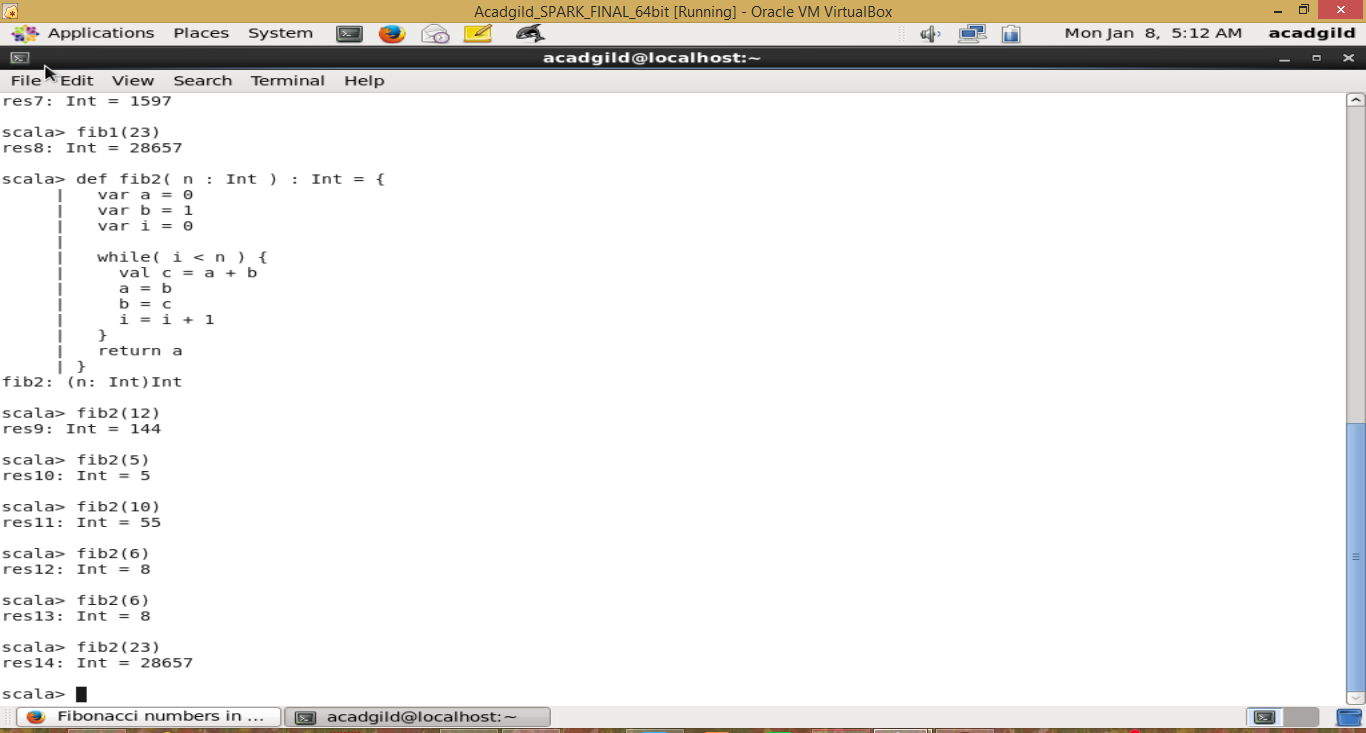
*scala> fib1(17)*

*res7: Int = 1597*

*scala> fib1(23)*

*res8: Int = 28657*

*scala>*



○ Write the function using recursion

scala> def fib2( n : Int ) : Int = {

| var a = 0

| var b = 1

| var i = 0

|

| while( i < n ) {

| val c = a + b

| a = b

| b = c

| i = i + 1

| }

| return a

| }

fib2: (n: Int)Int

scala> fib2(12)

res9: Int = 144

scala> fib2(5)

res10: Int = 5

scala> fib2(10)

res11: Int = 55

scala> fib2(6)

res12: Int = 8

scala> fib2(6)

res13: Int = 8

scala> fib2(23)

res14: Int = 28657

