ASSIGNMENT 13.3

Problem Statement:

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

Solution:

Finding square root of number using Babylonian method:

Here is the code snippet I have written in Scala to get this done:

```
object SquareRoot {

  def main(args: Array[String]) {

    val input = 50
    val result = getSquareRoot(input)
    println("Square root of given number" + input + "is " + result)

  }

  getSquareRoot(number: Double): Double = {

    val x = number
    val y = 1
    val e = 0.000001
    while((x-y) > e) {
      x = (x + y) / 2
      y = n / x
    }
    x
  }
}
```

Output:

Square root of given number 50 is 7.07106