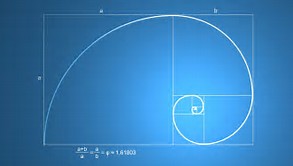
Session 13

Assignment 3

|  |  |
| --- | --- |
| **Prepared For:** | AcadGild |
|  |  |
| **Document Approval:** | **AcadGild** |
|  |  |
|  |  |
|  |  |
|  |  |
| **Project Title:** | Session 13 – Assignment 3 |
|  |  |
| **Prepared By:** | Duncan Burgess |
|  |  |
|  | dburgess@duncb.com |
|  |  |
| **Primary Engineer:** | Duncan Burgess |
|  |  |
| **Document Reference:** | **Session 12 – Assignment 3** |
|  |  |
| **Start Date:** | 10/10/2017 |
|  |  |
|  |  |



# 

# Contents

[Contents 2](#_Toc495599686)

[Change History 3](#_Toc495599687)

[1. Problem Statement 4](#_Toc495599688)

[2. Solution 4](#_Toc495599689)

# Change History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Revision** | **Date** | **Authored By** | **Authorised By** | **Sections Affected** | **Reason for Change** |
| Rev 01 | 12/10/2017 | Duncan Burgess |  | All | Initial release. |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 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

# Solution

**Code created**

***object*** *Babylonian {*

*println("Welcome to the Scala worksheet") //> Welcome to the Scala worksheet*

***def*** *sqrt1(n1:Int): Float = {*

***var*** *a: Float = n1*

***var*** *b: Float = 1*

***val*** *e:Double = 0.000001*

***while*** *(a - b >e) {*

*a = (a + b) / 2*

*b = n1/a*

*}*

*println ("The square root is ")*

*a*

*} //> sqrt1: (n1: Int)Float*

*Sqrt() //--- Enter value*

}

**Various results**

*sqrt1(99) //> The square root is*

*//| res0: Float = 9.949875*

*sqrt1(81) //> The square root is*

*//| res0: Float = 9.0*