Alexander Laudino

CSC-162-IN1

Dr. Farrett

Lab Assignment 4 – Ackermann Function Reference Documents

**Requirements**

Requirements for AckermannFunction

|  |  |
| --- | --- |
| Requirement | Priority |
| Method to return Ackermann function value | High |
| Display result for given parameters | High |

Requirements for DemoAckermannFunction

|  |  |
| --- | --- |
| Requirement | Priority |
| List containing input argument for method | High |
| Create new AckermannFunction object | High |
| Call method to display result | High |

**Pseudo-code**

This program demonstrates the Ackermann’s function.

DemoAckermannFunction

Begin

1. Initialize list of non-negative integer pairs
2. Create new AckermannFunction object as demo
3. Call demo.printAckermann(list)

End

The methods for Ackermann function and to display results

AckermannFunction

Begin ackermann(int m, int n)

1. If (m == 0)
2. Return n + 1
3. Else if (m > 0 and n == 0)
4. Return ackermann(m – 1, 1)
5. Else if (m > 0 and n > 0)
6. Return ackermann(m – 1, ackermann(m, n – 1)
7. Else
8. Return n + 1

End

Begin printAckermann(int[][] list)

1. Int c0 = 0
2. Int c1 = 1
3. For row in list
4. Print “Ackermann function value for (“ + list[row][c0] + “, “ + list[row][c1] + “): “ + ackermann(list[row][c0], list[row][c1])

End

|  |
| --- |
| PalindromeDetector |
|  |
| -ackermann(m : int, n : int) : int  +printAckermann(list : int[][]) |

**UML**

|  |
| --- |
| DemoAckermann |
| +list : int[][] |
|  |