**Student name: Thilan Maduranga Bentharage**

**Student ID: 10656631**

**Pseudocode for Assignment 1**

Import random package

Define system constant values

Define system requirement constants

Define show\_question\_answer(challenging=false) function with challenged parameter

Call to select\_question\_no method and store value in question for question type identification

If it is a challenging question

Double directory min and max values

If question is 1

Call to random\_list function with the parameters to get random generated array and store in my\_list array

Print instruction and my\_list

Get user answer and store it to answer param

If user answer is correct

Print Correct!

Increase score and correct\_answer by 1

Else

Print incorrect message and show correct answer

If question 2

Call to random\_list function with the parameters to get random generated array and store in my\_list array

Print instruction and my\_list

Get user answer and store it to answer param

If user answer is correct

Print Correct!

Increase score and correct\_answer by 1

Else

Print incorrect message and show correct answer

If question 3

Call to random\_list function with the parameters to get random generated array and store in my\_list array

Print instruction and my\_list

Get user answer and store it to answer param

If user answer is correct

Print Correct!

Increase score and correct\_answer by 1

Else

Print incorrect message and show correct answer

Else.

Call to random\_list function with the parameters to get random generated array and store in my\_list array

Print nstruction and my\_list

Get user answer and store it to answer param

Define select\_requirement(parameter) function

Create empty my\_options dictionary

If parameter is e

Assign my\_options to requirement directory

If parameter is m

Assign my\_options to requirement directory

If parameter is h

Assign my\_option to requirement directory

Return my\_options directory

Define random\_list(qty, min, max) function

Return randomly generated list under the requirements given

Define select\_question\_no function

Return randomly generated number which is between 1 to 4

Show welcome message

Create difficult parameter

Endless loop:

Prompt and assign user input

If user entered e, m or h:

Assign user input to difficult parameter

Stop loop

Else:

Print invalid choice

Call to select\_requirements(difficult parameter) to get required dictionary

Assign score, correct\_answers and i parameters to 0

Loop until question counts are met

Print question number

Call to show\_question\_answer mthod and check the response

If it is True

Increase score and correct\_answer by 1

Increase i

Print challenging question

Call to show\_question\_asnwer method with challenging question parameter True

Print test completed message and required details in the programme