Name: Lab8\_Knowlton

Author: Matthew Knowlton

Lab Number: 8

Date: 28 March 2022

BEGIN main

1. DECLARE scanner to capture user input
2. PRINT “Please input an integer: “
3. DECLARE userNumber using the next integer input from the scanner
4. DISPLAY “Number: userNumber“
5. DISPLAY “Binary: “
6. CALCULATE the result of binary using userNumber as the parameter and display it.
7. DISPLAY “userNumber Raised to 3: “
8. CALCULATE the result of power using userNumber and 3 as parameters and display it.
9. DISPLAY “Descending Order: “
10. CALCULATE the result of descending using userNumber as the parameter and display it.
11. DISPLAY “Sum: “
12. CALCULATE the result of sum using userNumber as the parameter and display it.
13. CLOSE scanner

END

BEGIN binary(int userNumber)

1. IF userNumber is equal to 0
   1. RETURN an empty string
2. ELSE
   1. CALCULATE the result of binary using userNumber divded by 2 as the parameter
   2. CALCULATE userNumber modulated 2
   3. RETURN the result of binary concatenated with the result of userNumber modulated 2
3. ENDIF

END

BEGIN power(int userNumber, int basePower)

1. IF basePower is equal to 0
   1. RETURN 1
2. ELSE
   1. CALCULATE the result of power using userNumber and basePower minus 1 as the parameters
   2. RETURN the userNumber multiplied by the result of power
3. ENDIF

END

BEGIN descending(int userNumber)

1. IF userNumber is equal to 0
   1. RETURN 1
2. ELSE
   1. CALCULATE the result of descending using userNumber minus 1 as the parameter
   2. RETURN the userNumber concatenated with the result of descending
3. ENDIF

END

BEGIN sum(int userNumber)

1. IF userNumber is equal to 0
   1. RETURN 0
2. ELSE
   1. CALCULATE the result of sum using userNumber divided by 10
   2. CALCULATE the result of userNumber modulated 10
   3. RETURN the result of sum plus the result of userNumber modulated 10.
3. ENDIF

END

TRACING n=9

DECLARE scanner = new Scanner(System.in)

PRINT “Please input an integer: ”

DECLARE userNumber = 9

PRINT “Number: 9”

PRINT “Binary: “

CALL binary(9)

RETURN binary(4) + “1”

RETURN binary(2) + “0”

RETURN binary(1) + “0”

RETURN binary(0) + “1”

RETURN “”

RETURN “1”

RETURN “0” + “1”

RETURN “0” + “01”

RETURN “1” + “001”

RETURN “1001”

PRINT “1001”

PRINT “9 Raised to 3: ”

CALL power(9, 3)

RETURN power(9, 2) \* 9

RETURN power(9,1) \* 9

RETURN power(9,0) \* 9

RETURN 1

RETURN 1 \* 9

RETURN 9 \* 9

RETURN 81 \* 9

RETURN 729

PRINT “729”

PRINT “Descending Order: “

CALL descending(9)

RETURN 9 + descending(8)

RETURN 8 + descending(7)

RETURN 7 + descending(6)

RETURN 6 + descending(5)

RETURN 5 + descending(4)

RETURN 4 + descending(3)

RETURN 3 + descending(2)

RETURN 2 + descending(1)

RETURN 1 + descending(0)

RETURN “”

RETURN “1 ” + “”

RETURN “2 “ + “1 “

RETURN “3 “ + “2 1 “

RETURN “4 “ + “3 2 1 “

RETURN “5 “ + “4 3 2 1 “

RETURN “6 ” + “5 4 3 2 1 “

RETURN “7 “ + “6 5 4 3 2 1 “

RETURN “8 “+ “7 6 5 4 3 2 1 “

RETURN “9 “ + “8 7 6 5 4 3 2 1 “

RETURN “9 8 7 6 5 4 3 2 1 “

PRINT “9 8 7 6 5 4 3 2 1 “

PRINT “Sum: “

CALL sum(9)

RETURN sum(0) + 9

RETURN 0

RETURN 0 + 9

RETURN 9

PRINT “9”

CLOSE scanner