CS4051 Fundamentals of Computing

Coursework

Numerical Data Program with Report

STUDENT NAME: DAN-MARIUS BRADEA

STUDENT ID: 20044497

STUDENT EMAIL: DAB1108@MY.LONDONMET.AC.UK

SUBMITED TO: Svetla Hubenova: s.hubenova@londonmet.ac.uk

Ramzi Djemai : r.djemai@londonmet.ac.uk

DATE: 09/05/2022

Contents

[Task 1. 3](#_Toc103018620)

[Construct a model of a byte adder based on the bit adder 3](#_Toc103018621)

[Task 2. 3](#_Toc103018622)

[Specify an algorithm for integer addition based on binary operation 3](#_Toc103018623)

[Task 3. 4](#_Toc103018624)

[Select suitable data structures to represent the information 4](#_Toc103018625)

[Task 4. 4](#_Toc103018626)

[Create a program in Python 3 which implement the model of the adder 4](#_Toc103018627)

[Task 5. 4](#_Toc103018628)

[Describe the program 4](#_Toc103018629)

[Task 6. 4](#_Toc103018630)

[Test the program with some sample data to demonstrate its behaviour 4](#_Toc103018631)

[Test 1. 4](#_Toc103018632)

[Test 2. 5](#_Toc103018633)

[Conclusion 5](#_Toc103018634)

# Task 1.

### Construct a model of a byte adder based on the bit adder

Diagram

Description automatically generated

# Task 2.

### Specify an algorithm for integer addition based on binary operation

Taking in to account the logic gates given to us by the tutor the basic design of the bit adder should be like it follows

Step 1. Prompt user through a message to input two positive integers within a specific range

Step 2. Make sure that the input is valid (positive, within the specified range)

Step 3. Take integers inputted by the user and store them in their unique variables

Step 4. Transform integers in to binary notation using bin()

Step 5. Remove “0b” from the beginning of the binary notation of the numbers

Step 6. Split the binary notation and store the digits in to their separate corresponding list

Step 7. Make sure that both lists are 8 characters long

Step 8. After conditions are meet

Step 9. Define a function that corresponds to the diagram of the bit adder given to us by the tutor in the coursework requirements

Step 10. Take the last digit from the list corresponding to the first input and the last digit from the list corresponding to the second input and pass them along with the carry\_in through the logic gates as in the code snippet below

def full\_bit\_adder(upper, lower, carry\_in):# In the beginning the carry\_in is “0”

    output1 = AND(upper, lower)

    output2 = XOR(upper, lower)

    output3 = AND(output2, carry\_in)

    sum = XOR(output2, carry\_in)

    carry\_out = OR(output1, output3)

    return sum, carry\_out

# Task 3.

### Select suitable data structures to represent the information

Primitive data types suitable for this project:

* Integer
* String

Non-primitive data type suitable for this project:

* List

# Task 4.

### Create a program in Python 3 which implement the model of the adder

The program is attached to the submission as instructed

# Task 5.

### Describe the program

The program is described in the comments of the program in as much detail as I am capable of.

# Task 6.

### Test the program with some sample data to demonstrate its behaviour

## Test 1.

In the flowing test the wrong input is entered 3 times and all these times the program asks for an input till the input is within the range needed

After the input satisfies the requirements the program moves to the second input.

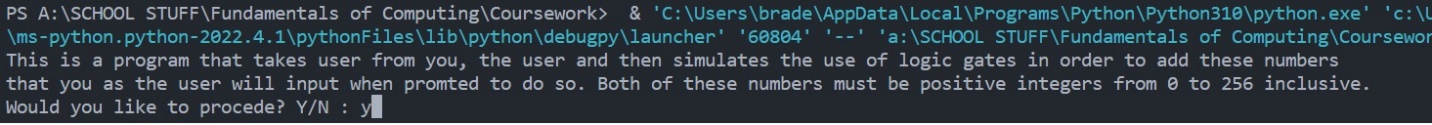
After the second input is validated the program informs the user of the binary transformation and prints the lists corresponding to each input.

Even if the addition is done and result is correct there is information missing.

Text

Description automatically generated

## Test 2.

The first test has failed to provide important information to the user from the beginning and the option to go ahead with the program or not to go ahead.

In this test the user is prompted with a message explaining what the program does and what s expected of him, and he is given the choice to go ahead or quit. If the user choses no “N”

Graphical user interface

Description automatically generated

If the user choses “Y” then the program moves to collect the required input.

# Conclusion

In conclusion, this project has really made me scratch my head and realise that failure and practice are the only ways in which one can learn to program. If the purpose of this coursework was for people to understand step by step how programs work in the background of everyday things, then it succeeded in making me understand that.