CMPT 120 - Assignment 1

Due Date: Thursday, 23 February, 11:59 PM

Given the code below:

```
# Main Program
n1 = int(input("Enter n1: "))
n2 = int(input("Enter n2: "))
ans = n1 + n2

str1 = decimalToBinary(n1)
print("The binary representation of", n1, "is", str1)

str2 = decimalToBinary(n2)
print("The binary representation of", n2, "is", str2)

str3 = binaryAddition(str1, str2)
print("The binary addition of", n1, "and", n2, "is", str3)

n3 = binaryToDecimal(str3)
print("Converting the binary to decimal gives", n3)

if ans == n3:
	print("Since", ans, "==", n3, ", it seems we did good job.")
else:
	print("Since", ans, "!=", n3, ", something went wrong.")
```

define the three functions **`decimalToBinary`**, **`binaryAddition`**, and **`binaryToDecimal`** so that the program will be able to add binary numbers. The description of each function is as follows:

- **`def decimalToBinary(n):`**
  - This function takes a positive integer **`n > 0`** as an argument and returns a string of **integers** where each character of the string is a binary digit, 0 or 1 obtained by the conversion of the integer argument **n** to binary.

    You are not allowed to use the built-in function `bin()`.

    For example,

    `decimalToBinary(3)` → **`"11"`**
    `decimalToBinary(123)` → **`"1111011"`**
    `decimalToBinary(2312)` → **`"100100001000"`**

- **`def binaryAddition(str1, str2):`**

  - This function takes two string arguments `str1` and `str2` representing two binary numbers and returns a string of integers whose characters are obtained by the addition of the two binary numbers in binary addition format.

  - How to add two binary numbers? Binary addition is like decimal addition, except that it carries on a value of 2 instead of a value of 10.

    ```
    0 + 0 = 0
    1 + 0 = 1
    0 + 1 = 1
    1 + 1 = 10 (this is 0 carry 1)
    1 + 1 + 1 = 11 (this is 1 carry 1)
    ```

    To add two binary numbers, we simply add together the corresponding digits in each binary number from right to left and if the sum is greater than what fits in a single digit, we carry a 1 into the next column.

    For example,

    ```
      1 1 1 1   1      ← carries
        1 1 0 1 0 1
    +     1 1 1 0 1
    ---------------
      1 0 1 0 0 1 0
    ```

    We start by the first column from the right:

    1) First column: `1 + 1 = 0` (with carry 1)
    2) Second column: `0 + 0 + 1 (carried) = 1` (no carry)
    3) Third column: `1 + 1 + no carry = 0` (carry 1)
    4) Fourth column: `0 + 1 + 1 (carried) = 0` (carry 1)
    5) Fifth column: `1 + 1 + 1 (carried) = 1` (carry 1)
    6) Sixth column: `1 + 1 (carried) = 0` (carry 1)
    7) Seventh column: `1 (carried)`

    `binaryAddition("110101", "11101")` → **`"1010010"`**

`binaryAddition("110101", "11101")` → **`"1010010"`**

- **`def binaryToDecimal(binStr):`**
    - This function takes a string **`binStr`** representing a binary number and converts it to decimal and returns the decimal number.

      For example,
      ```
      binaryToDecimal("10") → 2
      binaryToDecimal("10100") → 20
      binaryToDecimal("101110110010") → 2994
      binaryToDecimal("1111101110110000") → 64432
      ```

Submission Format

You are required to submit your program online through Moodle. You will find a submission button for **Assignment 1** on **Moodle under Assignment Folder**.

I will not accept any submission by email.

Marking

A non working program will automatically get zero. A program that works but doesn't give right output or gives partial right output will lose marks depending how severe its shortcoming is.