(CSCI 365) Project Two: A Binary Number Calculator Updated

**Objectives:**

1. Demonstrate basic programming logic
2. Demonstrate how to convert between integers and their binary representation in two’s complement format
3. Demonstrate how to add two binary numbers that are in two’s complement format

**Problem Description:**

Write a C/C++ program that simulate a menu based binary number calculator. This calculate shall have the following five functionalities:

1. Covert a binary string in two’s complement format to corresponding integers
2. Convert an integer to its binary representation in two’s complement format
3. Add two binary numbers, both numbers are represented as a string of 0s and 1s
4. Provide sign extension for a binary number
5. Provide two’s complement for a binary nunber

This is an update of Project One. To reduce student work load, a start file CSCIProjTwoHandout.cpp is given. In this file, the structure of the program has been established. The students only need to implement the following five functions:

1. int binary\_to\_decimal\_signed(string b);

// precondition: b is a string that consists of only 0s and 1s

// postcondition: the decimal integer that is represented by b

1. string decimal\_to\_binary\_signed(int n);

// precondition: n is an integer

// postcondition: n’s binary representation is returned as a string of 0s and 1s

1. string add\_binaries\_signed(string b1, string b2);

// precondition: b1 and b2 are strings that consists of 0s and 1s, i.e. b1 and b2 are binary

// two’s complement representations of two integers

// postcondition: the sum of b1 and b2 is returned. For instance, if b1 = “11”, b2 = “01”, // then the return value is “100”

1. string signed\_extension(string b);

// precondition: s is a string that consists of only 0s and 1s that is at most 16 bits

// postcondition: a 16 bit string has been returned as signed extension of s. For instane,

// if s = "0101" then return value will be "00000000000000000101" total 12

// 0s are added in front of s

1. string twos\_complement(string s);

// precondition: s is a string that consists of only 0s and 1s

// postcondition: two's complement of s is returned as an 16 bits binary integer. For

// instance, if s = "1101", then return value will be "1111111111111101"

The following functionality has been provided. Please notice that three functions from Project One are provided. The idea is your new function will somehow call corresponding one in these three functions to do the work:

1. main function which presents the execution logic of the whole program
2. void menu(); which display the menu of this binary calculator
3. int binary\_to\_decimal(string b);

// precondition: b is a string that consists of only 0s and 1s

// postcondition: the positive decimal integer that is represented by b

1. string decimal\_to\_binary(int n);

// precondition: n is a positive integer

// postcondition: n’s binary representation is returned as a string of 0s and 1s

1. string add\_binaries(string b1, string b2);

// precondition: b1 and b2 are strings that consists of 0s and 1s, i.e. b1 and b2 are binary

// representations of two positive integers

// postcondition: the sum of b1 and b2 is returned. For instance, if b1 = “11”, b2 = “01”, // then the return value is “100”

1. bool isBinary(string s); which returns true if the given string s consists of only 0s and 1s; false otherwise
2. int grade(); which returns an integer that represents the student’s grade of this projects.
3. bool test\_binary\_to\_decimal() which returns true if the student’s implementation of binary\_to\_decimal function is correct; false otherwise
4. bool test\_decimal\_to\_binary() which returns true if the student’s implementation of decimal\_to\_binary function is correct; false otherwise
5. bool test\_add\_binaries which returns true if the student’s implementation of add\_binaries function is correct; false otherwise

**Work Process:**

Student shall download the CSCI365ProjTwoHandout.cpp file, saved as YourNameCSCI365ProjTwo.cpp, and add it to his/her C++ project. Please notice, at this point, your project shall be able to run. However, your project will not pass any test. i.e. if you choose option 6 from menu, it will shows that you get 0 out of 10.

Student shall NOT change any name of the declared functions or the given implemented functions. Student shall only implement the bodies of three functions that need to be implemented.

Please see sample run at the end of this file.

**Due Date:**

It is specified in syllabus and is announced on Blackboard

**Sample Run:** The highlighted ones are user’s input. The red part is explanation of result, which doesn’t show up in sample run output

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 1

Enter a binary string: 1001101

Its decimal value is: -51 (the two’s complement of 1001101 is 0110011 which is positive 51. So 1001101 is -51)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 2

Enter an integer: -51

Its binary representation is: 1111111111001101 (51 is 0110011. So -51 is 1001101. Use sign extension to sixteen bits, it is 1111111111001101)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 3

Enter two binary numbers, separated by white space: 1001101

01101

The sum is: 1111111111011010 (both numbers get sign extension as 1111111111001101 and 0000000000001101. Then add them bits by bits with carry to get 1111111111011010)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 4

Enter a binary number: 1001101

Its signed extension to 16 bits is: 1111111111001101 (In fact, this one must be right before the first three options can be right)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 5

Enter a binary number: 1001101

Its two's complement is: 0000000000110011 (In fact, this one must be right before the first three options can be right)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 6

binary\_to\_decimal\_signed function pass the test

decimal\_to\_binary\_signed function passed the test

add\_binaries\_signed function passed the test

sign\_extension function passed the test

twos\_complement function passed the test

If you turn in your project on blackboard now, you will get 8 out of 10

Your instructor will decide if one-two more points will be added or not based on your program style, such as good commnets (1 points) and good efficiency (1 point)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Menu \*

\* 1. Binary to Decimal \*

\* 2. Decinal to Binary \*

\* 3. Add two Binaries \*

\* 4. Signed extension \*

\* 5. Two's complement \*

\* 6. Grade \*

\* 7. Quit \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice: 7

Thanks for using binary calculator program. Good-bye

Program ended with exit code: 0