CS244 Data Structure
Programming Assignment 3
(100 points)
No late submission will be allowed!

Objective:

- Practice on using stack.

Problem Statement

In the previous class, we implement a Stack ourselves. It is time to test it! In this programming assignment, your task is to write a program that *use the stack (you can use the one implemented in the class)* to convert a Number from Binary to Decimal

i.e., convert 1001101 to decimal:

$$1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$
$$= 64 + 0 + 0 + 8 + 4 + 0 + 1 = 77$$

Weight of each bit: 6 5 4 3 2 1 0
Corresponding bit: 1 0 0 1 1 0 1

To convert we need to consider:

1. Weight of each bit
2. Value of each bit, i.e., 0 or 1
3. Use module operation to extract bits
4. Use stack to hold temporary status or values to be summed.

Your program should:

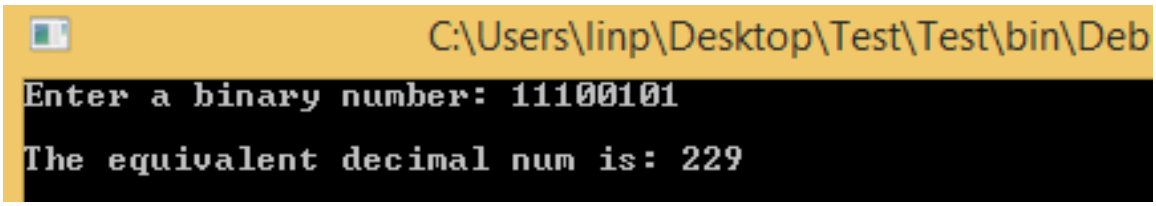
1. Convert any give binary number into an equivalent decimal number.
2. Check for invalid use inputs.
3. Write a class with the convert function.
4. Separate header and implementation files.
5. ***MUST implement a stack (or use the one implemented in class) for this assignment.***

Testing:

Test your program for: 11000101, 10101010,
11111111, 10000000, 1111100000.

Your program must compile and execute without any errors or warnings.

Sample output:



Submission Guidelines:

To receive full credit, you must follow these guidelines

1. All code must include comments, be properly indented and use descriptive variable names where appropriate.
2. Compress the entire project folder with solution directory to a compressed file and *submit this to the appropriate dropdox on D2L*
3. After submission, double check to make SURE your submit the proper files, ***YOU WILL NOT HAVE A CHANGE TO FIX THIS AFTER THE DUE DATE!!***
   Please ask for help if you need it.

Grading:

1. If your program does not compile and lack efforts (i.e., lack comments and function implements), you receive 0.
2. If your program does not compile but shows significant efforts in code, you receive at most 30% of the total credit.
3. If you program can compile but missing functionalities (i.e., nice interface), you receive about 60% of the total credit. You may receive more than 60% according to the following rubrics.

	Points	Deducted Points
Correctness	80	
1. Program repeatedly display the main menu	10	
2. Program displays the correct output	30	
3. Program check for invalid inputs	10	
4. Program can convert large binary number (20 bits)	30	
Style	20	
Lay out your program in a readable fashion and user-friendliness	5	
Include comments for each function	15	
Your Score		