CS244 Data Structure
Programming Assignment 2
(100 points)
No late submission will be allowed!

Objective:

- Practice on recursive algorithm design.

Problem Statement

Two more number systems, octal (base 8) and hexadecimal (base 16), are of interest to computer scientists. In fact, in C++, you can instruct the computer to store a number in octal or hexadecimal.

The digits in the octal number system are 0, 1, 2, 3, 4, 5, 6, and 7. The digits in the hexadecimal number system are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. So A in hexadecimal is 10 in decimal, B in hexadecimal is 11 in decimal, and so on.

The algorithm to convert a positive decimal number into an equivalent number in octal (or hexadecimal) is explained in the following. Here, we divide the decimal number by 8 (for octal) and by 16 (for hexadecimal). The remainder of the division is the rightmost digit of the number system of our interest (either octal or hexadecimal). The quotient will be divided again by 8 or 16 and the remainder resulted from the division is the second rightmost digit. We repeat this division process until quotient becomes zero.

Suppose a_b represents the number a to the base b. For example, 75_{10} means 75 to the base 10 (that is decimal), and 83_{16} means 83 to the base 16 (that is, hexadecimal). Then:

$753_{10} = 1361_8$
$753_{10} = 2F1_{16}$

The method of converting a decimal number to base or 8, or 16 can be extended to any arbitrary base. Suppose you want to convert a decimal number n into an equivalent number in base b, where b is between 2 and 36. You then divide the decimal number n by b as in the algorithm for converting decimal to binary.

Note that the digits in, say base 20, are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, and J.

Write a program that uses a ***recursive function*** to convert a number in decimal to a given base b, where b is between 2 and 36. Your program should prompt the user to enter the number in decimal and in the desired base.

Test your program on the following data:
9098 and base 20
692 and base 2
753 and base 16

```
Enter a nonnegative integer: 9098
Enter the desired base between 2 and 36: 20
9098 to the base 20 = 12EI
```

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 any base in [2-36]	30	
Style	20	
Lay out your program in a readable fashion and user-friendliness	5	
Include comments for each function	10	
Provide separate .cpp files and header files for class(es)	5	
Your Score		