1. Project Description of 1

Assignment 1 is to repeat the function by 'collage guessing' until the integer is input and the integer becomes 1.

Before you create a program, you should understand what 'collateral guess, 'hail number' is, and what functions are used in programming.

You need to understand the use of the while, for, and do-while functions to calculate iteratively with one value at the most. You also need to know how to use the if function to execute the function when you have a condition.

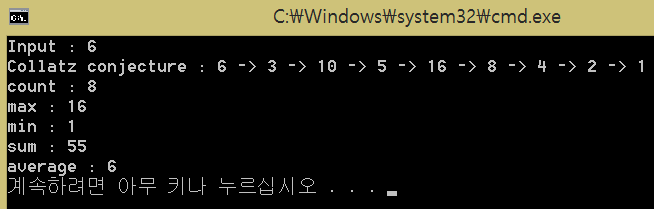
In the assignment, we had to output the sum, minimum value, maximum value, calculation frequency, average value of the numbers. When the program is executed, it receives the number and executes the function.

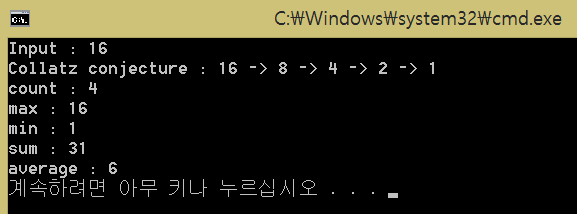
When the number 1 is received, the function is designed to be terminated immediately.

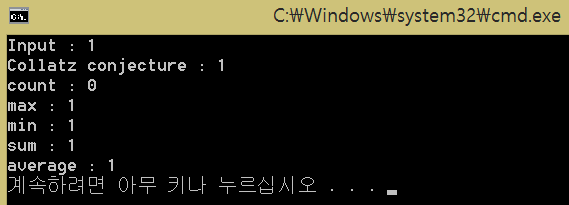
If the number is not 1, the calculation must be performed. Therefore, it shows the number of times the calculation is performed while outputting "->".   
If the input number is an even number, it divides it by 2 and stores it.   
Since the minimum value is a value that can be output only when dividing, the minimum value is saved when the number that was input first becomes smaller than the value divided by 2.   
In the same way, multiply Hole 3 by 1, add 1, and store the largest value as the maximum value.

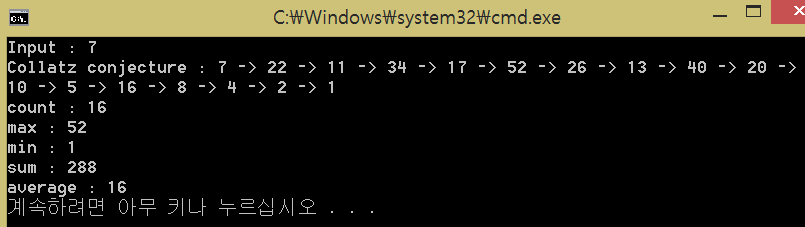
The totals are executed every time the calculation is made, and the number of calculations is also increased by one, and the average value is calculated as the derived values.

1. Action Results









1. Review

The most troubling part of creating the program was to print "->".

The problem is that, last 1 is output and "->" is output one more time.

So, using debugging, we found problems with outputting or not outputting one more time, and resolved the problem by repositioning the function.

The number of times, the maximum value, the minimum value, the average value, and the conditional statement were used to obtain the value without difficulty.

1. Project Description of 2

The second task is that, using conditional loop since we were freshmen.

In this assignment, we will learn the basics of C++ by mastering loops by drawing stars with Your Mark.

Input Your mark, and Input N that is length of one side of star. N is an odd number between 3 and 11.

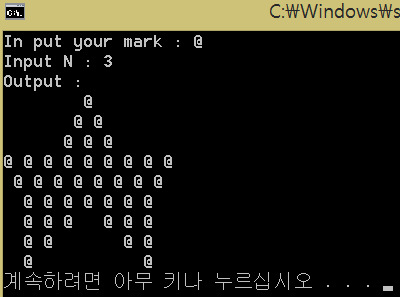
So, I design the program that creates a star shape by receiving the mark you want and entering a number that matches the range.

Exception handling is designed to accept only odd integers between 3 and 11, If the range does not fit, we designed the integer to be re-entered.

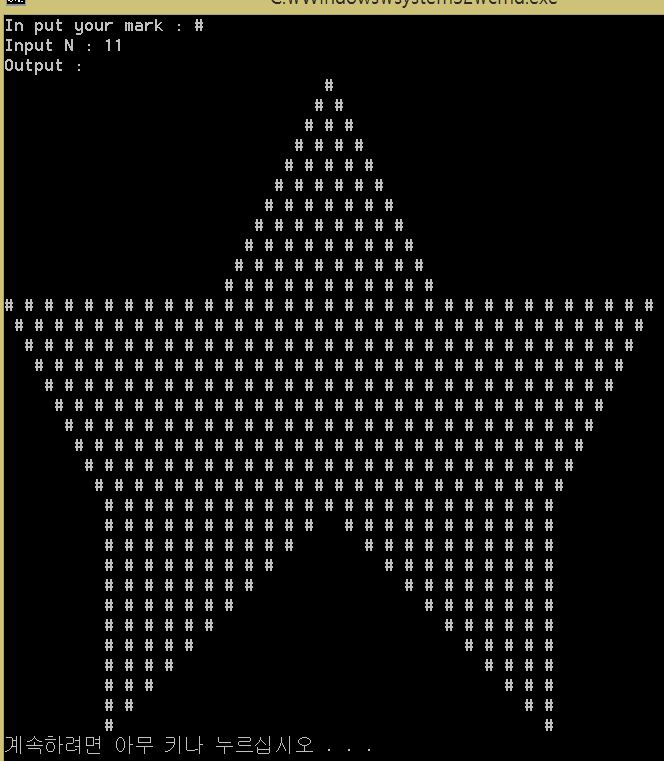
I design the code divides the star into three parts to find the regularity.

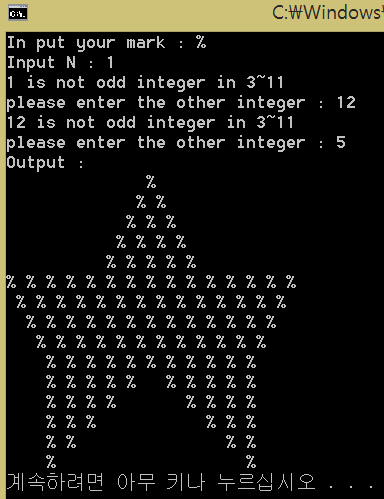
I used letters and blanks appropriately to make it look like the stars in the assignment.

1. Action Results









1. Review

It took me a while to think about splitting it into three parts when I made a star.

Because I must think of characters with Blank, I thought a lot when I gave them the conditions.

I just used blank because I did not get a shape when I received only stars.

The difficulty was not the hardest, but the overall time was the program I had the most.

Exception handling is easily solved with a while statement.

There are a lot of situations as well as the handling of exceptions in the task of exception handling, but it seems to be a bit lacking in implementation yet.

1. Project of description of 3

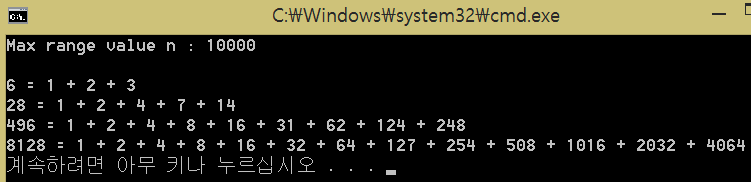
Task 3 is a program that receives a range and finds a 'perfect number' within that range.

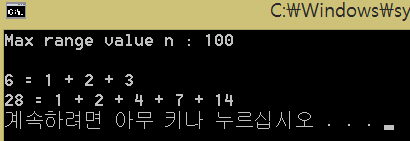
If you understand what the 'Perfect number' is before you start the project, you can print out the results using the loop as appropriate.

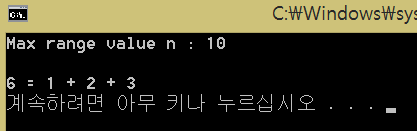
First, 'Perfect number' means any number A, and if the sum of the digits except the A number itself is A, the number is 'Perfect number'.

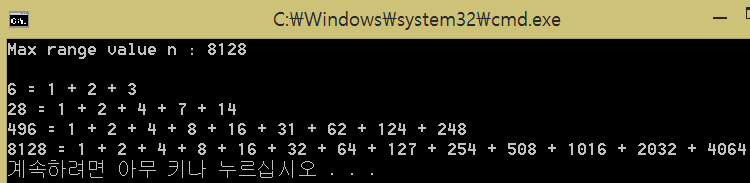
In the process of executing the function, we used three loops to derive the 'Perfect number' through the function and input the addition process as shown in the task result value.   
If the 'Perfect number' is found by comparing the values ​​of the whole range entered and the 'Perfect number' is found, based on the values, we have implemented a repetitive function to calculate the divisor so that when the divisor is found, it can be used to locate another Perfect number.

1. Action Results









1. Review

While programming the third problem, it was easy to get only the "Perfect number", but I was worried about the output of the divisors.   
At first, I was trying to design a function that outputs the value of the divisor using only the variable that outputs the 'Perfect number' value.

But We feel that there are not enough variables to calculate, so using the method, when you start the function, it gives the first range, continues to increment the variable, compares it to the range you entered.

Finally, we solved the problem by using the "Perfect number" method to calculate the divisor.

When you find the divisor, you get the 'Perfect number', save 1 to other variables, increment it by 1, compare it with the 'Perfect number', and if it is the divisor, output it.

We solved this by designing the function If the variable is equal to 'Perfect number' to be terminated.

1. Project Description of 4

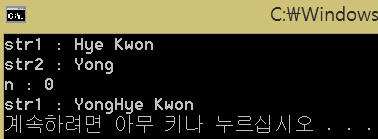
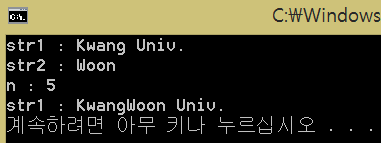
It is a program that receives the first string, receives the second string, and inserts the second string into the space of inputted number at first string.   
The method of insert a string into another string is that.

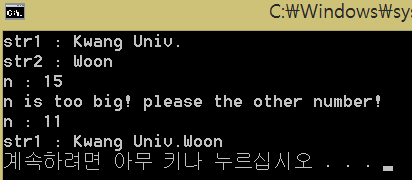
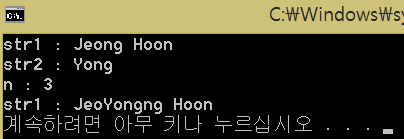
first, the string of in first string that entered after number of you entering save another sub string.

Second, you can solve the assignment by typing the second string and then saving the character back.

In addition, we designed to receive the number again if the input number is larger than the length of the string.

1. Action Results





1. Review

This was one of the most difficult tasks for me. Regardless of the integer variable you usually write.

I did not get a good understanding of string functions, input functions, and characterization of strings in programming.

While executing the actual function, many errors occurred because of NULL character, Later, debugging will allow NULL characters to be placed at the end of the string and I have also tried to understand the function to include the Enter character by using the input function used in C language.

I first studied the concept of strings and designed the program again so that I could design the program much easier than before.

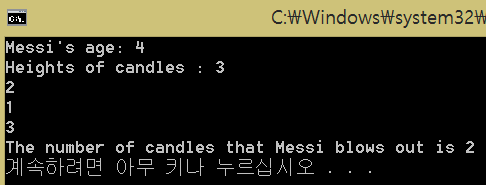
1. Project Description of 5

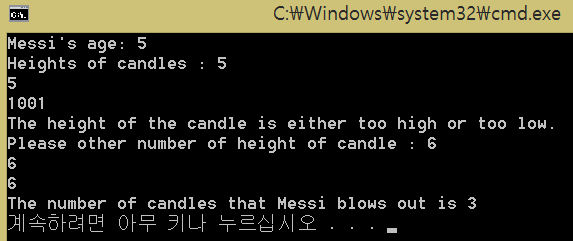
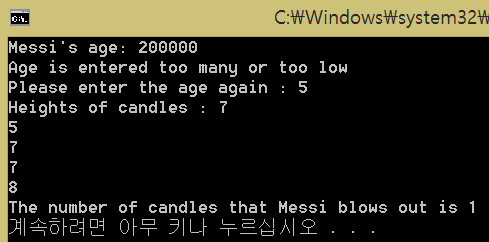
The first input receives the number, it is a program that outputs the number of the highest candles by inputting the number of candles as many as that number.

Exception handling requires that the first number of inputs be in the range 1 <= N <= 100000 and the height of the candle should also be in the range 1 <= N <= 1000.   
If the range is out of range, it is designed to receive the number again. We used the conditional statement and the loop statement as the used functions.

When the condition is satisfied while using while and Continue, it is designed to return to the beginning of the loop statement function from the place where the Continue was executed, so that the function is executed again so that the candle height can be inputted.

1. Action Results





1. Review

There were many difficulties in handling exceptions.   
As shown in the results of the priority task, when the space bar number is input, there is a problem that the output characters overlap and come out together.

The exception handling made it possible to re-enter the candle height when it was out of range If you receive a space bar, if the number exceeds 2, I have difficulty in designing because the message is duplicated.

It was easier to design an exception handling by asking the assistant through e-mail and accepting input by Enter.