Personal Assignment #2: Stacks and Queues Total = 60 points

Topic	Stack	Total = 30 points

Requirement 1: Your program should define the MAX_STACK_SIZE [2 points]

Requirement 2: Your program must implement push( ) operation [5 points]

Requirement 3: push( ) operation should return the following error message when the stack is full. "Scanning integers has stopped, the stack is full, I will pop out the phone number from the stack." [2 points]

Requirement 4: Your program must implement pop( ) operation [5 points]

Requirement 5: pop( ) operation should return the following error message when the stack is empty, "The Stack is now empty, continue scanning."

Requirement 6: pop( ) operation should return the message "Scanning is completed." When it has reached the end of the input string. [3 points]

Programming Exercise#1: You are given a string of numbers as input which contains a list of phone numbers. The input string contains several phoner numbers which are concatenated together as shown below.

0107538221701023456987010465389 87

This is makes it difficult to read to the phone numbers.

[Functional requirements of the program]

(a) You are required to write an array implementation of a stack program that separates the phone numbers. Your program must follow requirements 1 to requirement 5 mentioned above.

(b) The maximum stack size should be equal to the number of integers contained in one phone number.

(c) Your program should scan the numbers from left to right and push each integer into the stack. [5 points]

(d) When the Stack is full, all the integers in the stack should be popped out and stored in a separate variable called *phone number [i].* [5 points]

(e) Continue scanning to the right until the program reaches the last integer in the input file.

(f) Print the list of the separate phone numbers. [5 points]

** The implementation style is up to the student, but the student should only tackle the problem using the array implementation of the stack.

An example output should look like this.

```
Scanning integers has stopped, the stack is full, I will pop out the phone number from the stack.

phoneNumber1: 01075382217

The Stack is now empty, continue scanning.

Scanning integers has stopped, the stack is full, I will pop out the phone number from the stack.

phoneNumber2: 01023456987

The Stack is now empty, continue scanning.

Scanning integers has stopped, the stack is full, I will pop out the phone number from the stack.

phoneNumber3: 01046538987

Scanning is completed.

The list of phone numbers from the input string is.

01075382217

01023456987

01046538987
```

Documents to submit:

1. Screenshot for the console as shown in the example above
2. Name your program file as stack.c and submit it.

Topic	Queue	Total = 30 points

Using Queue, develop a program to manage patients waiting for treatment in hospitals. The conditions given below and the functionality of the program. Write the code to meet the requirements. Submit the code and report together.

[Functional requirements of the program]

① The program should first present the following message on the console that prompt for user input "Welcome to the Hospital care service. Please select the number of the function you want."

② The user selects 'Hospital Arrival', ' Proceed treatment'', and 'Service Cancellation" via a console.

③ Selecting ' Service Cancellation ' will terminate the program, otherwise it will continue to wait for the user's input.

④ The maximum number of people waiting is limited to five.

⑤ If you select "Hospital Arrival" and there are already five people waiting, you will print a message saying, "There is no space for new medical treatments, try again next time."

⑥ If you chose 'Hospital Arrival', and there are fewer than 5 people waiting, print out the current waiting number and prompt for "Waiting registration."

⑦ When the user selects "Waiting registration", the name of the waiting person is registered, and program returns to the initial console input screen.

⑧ When the user selects 'Proceed treatment', the name of the first customer who can be treated among the waiting people is displayed on the screen.

⑨ However, when the user selects 'Proceed treatment', and there is no patient waiting in the queue, a message should print 'No patient waiting'.

*** The waiting name is a string of up to 20 characters. If you enter the number of a function that is not presented on the screen, an error message is printed.*

Additional functions such as message and screen configuration provided on the user screen and checking the user's input should satisfy the above functions.

** Within the scope, the implementation style is up to the student. However, the input should proceed in the same way as in the case below.

Your Program sequence output should look like below. So, take screenshot of your console to show the example cases below.

Welcome to the Hospital care service. Please select the number of the function you want.

1. Hospital Arrival 2. Proceed treatment 3. Cancellation **[3 points]**

4

You entered the wrong number. --------- Welcome to the Hospital care service. Please select the number of the function you want. 1. Hospital Arrival 2. Proceed treatment 3. Cancellation **[5 points]**

1

The current waiting number is 0. Please select the number of the function you want from the following. 1. Waiting registration 2. Cancellation **[3 points]**

1

Please register the name of the waiting person: James **[2 points]**

Thank you and ..... Welcome to the Hospital care service. Please select the number of the function you want.

1. Hospital Arrival 2. Proceed treatment 3. Cancellation **[3 points]**

1

The current number of people waiting is 1. Please select the number of the function you want from the following. 1. Waiting registration 2. Cancellation **[7 points]**

1

Please register the name of the waiting person: Joy

Welcome to the Hospital care service. Please select the number of the function you want.

1. Hospital Arrival 2. Proceed treatment 3. Cancellation

** On top of the above program sequence, you should show that your program satisfies functional requirement #4. **[7 points]**

Documents to submit:

1. Screenshot for the console as shown in the example above
2. Name your program file as queue.c and submit it.