CIS 277 002HY

Professor Aljamal
Stacks and arrays programming project
Chaemin Chung

```
Program start
Creat a structure named “stack” to create the stack
    Declare an array of integers with a size of 10.
    Declare an integer variable, representing the top of the stack.
End of structure

Declare a prototype to check if the input valid(parameter: integer)
Declare a prototype to check if the stack is empty.(parameter: stack)
Declare a prototype to check if the stack is full.(parameter: stack)
Declare a prototype to push the integers into the stack(parameter: stack)
Declare a prototype to pop the integers from the stack(parameter: stack)
Declare a prototype to display the top of the stack(parameter: stack)
Declare a prototype to purge the stack(parameter: stack)
Declare a prototype to display the stack(parameter: stack)

Initialize main function
    Display the beginning title of the program.
    Initialize the integer variable “menuChoice” to get user input.
    Initialize the stack variable “stack1”
    Use stack purge function to purge stack1.

    Create a while loop
        Display a horizontal line
        Display the word “menu”
        Display option 1 “Push”
        Display option 2 “Pop”
        Display option 3 “Top”
        Display option 4 “Purge”
        Display option 5 “Display”
        Display option 6 “Exit”
        Display a horizontal line

        Display a prompt so user can select an option
        Get user input in menuChoice.
        Create a new line

        Create a switch statement for the menu
            Write case 1
            Enter case 1
    Display a title to indicate that the program is performing the "Push" operation
    Call function to push integer into stack
    Create a new line
    Break.
Close case 1

Write case 2
Enter case 2
    Display a title to indicate that the program is performing the "Pop" operation
    Call function to pop integer from stack
    Create a new line
    Break
Close case 2

Write case 3
Enter case 3
    Display a title to indicate that the program is performing the “Top” operation
    Call function to display top of the stack
    Create a new line
    Break
Close case 3

Write case 4
Enter case 4
    Display a title to indicate that the program is performing the “Purge” operation
    Call function to purge the stack
    Create a new line
    Break
Close case 4

Write case 5
Enter case 5
    Display a title to indicate that the program is performing the “Display” operation
    Call function to display the the entire stack
    Create a new line
    Break
Close case 5

Write case 6
Enter case 6
    Indicate the user the program is end
    Write the return statement
    Break
Close case 6
```

```
            Write default case
            Enter default case
                Indicate the user that an error occurred
                Prompt user to enter valid menu choice
            Break
            Close default case
        Close switch statement
    Close while loop
Close main function

Initialize the bool function that checks if the input valid.(parameter: int)
    If integer is less than 0 or bigger than 99 then
        Indicate the user the input is not valid.
        Return false
    Close the if statement brace
    Else
        Return true
Close the function

Initialize the bool function that check if the stack is empty(parameter: stack)
    If the top, a member of the structure ‘stack’, is -1 then
        Indicate the user the stack is empty
        Return true
    Close the if statement brace
    Else
        Return false
Close the function

Initialize the bool function that check if the stack is full(parameter: stack)
    If the top, a member of the structure ‘stack’, is 9 then
        Indicate the user the stack is full
        Return true
    Close the if statement brace
    Else
        Return false
Close the function

Initialize the void function that pushes the integers into the stack(parameter: stack)
    Initialize the integer variable i to push the stack
    If the stack is full then
        Indicate the user that the operation failed
    Close the if statement brace
    Else
```

```
        Display the message to ask user for a integer to push into the stack
        Get integer
        Call the function to check integer is valid and if it is valid then
            Increase 1 the value of top, a member of the structure ‘stack’
            Store the integer to a top of the stack
            Display entire stack
        Close if statement
    Close else statement
Close the function

Initialize the void function that pops the integers from the stack(parameter: stack)
    If the stack is empty then
        Indicate the user that the operation failed
    Close the if statement
    Else
        Display the the integer popped from the stack
        Decrease 1 the top, a member of the structure ‘stack’
        Display entire stack
    Close the else statement
Close the function

Initialize the void function that displays the top of the stack(parameter: stack)
    If the stack is empty
        Indicate the user that the operation failed
    Close the if statement
    Else
        Display the top of the stack
        Display the entire stack
    Clseo the else statement
Close the function

Initialize the void function that that purges the stack(parameter: stack)
    Set the top, a member of the structure ‘stack’ as -1
Close the function

Initialize the void function that displays the stack
    If the stack is not empty then
        Display the message indicating that the following output is the current stack status
        Create a for loop that iterates from 0 to the size of the current stack
            Print each element of the stack
        Close the for loop
    Close the if statement
    Display total number of elements in the stack and its maximum capacity.
Close the function
```