

**Data Structure Lab Project**

SUBMITTED TO

Mr. Masud Rabbani

Lecturer

Department of Computer Science and Engineering.

Daffodil International University.

SUBMITTED BY

Md. Anisur Rahman Rony

ID: 162-15-7880.

Course Code: CSE-135.

Department: Computer Science and Engineering.

Section: E.

Team Name: “RONY”.

Date of Submission: 11 April 2017.

**Project Title**

Making four basic functions of “calculator” using stack.

**Which have been used in this project?**

1. Stack.

2. Push() function.

3. Pop() function.

4. Dynamic memory allocation [using malloc() function].

5. Pointer etc.

**Steps to make this calculator**

1. First of all, a pointer has been created in the main function and maximum size of the stack has been defined

at first of the program.

2. Using these pointer and maximum size top and bottom

of stack has been assigned.

3. Two user defined functions push() and pop() have been created outside the main function.

4. In the main function a switch case has been used to perform four basic operations (Addition, subtraction, multiplication and divide) of calculator.

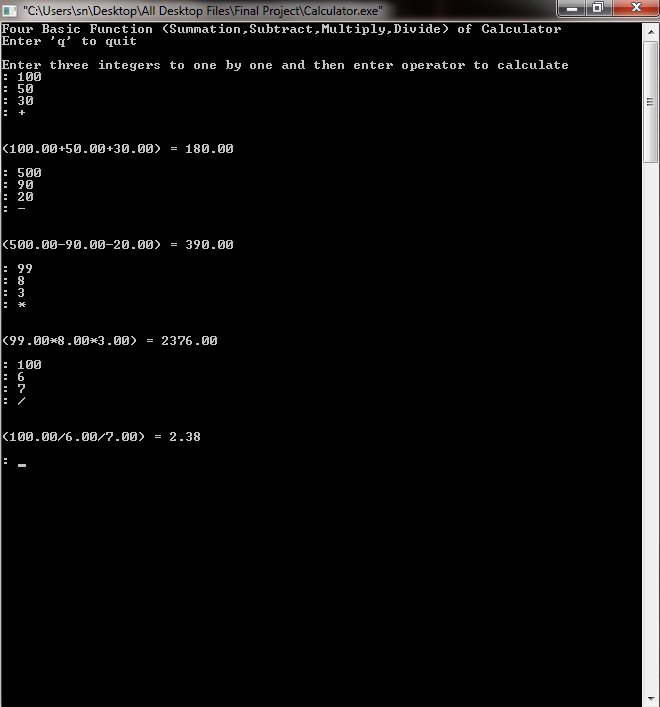
5. In every case inside switch case I have taken the values from pop() and push it to the push() function.

6. In this program I took three integers from the pop() function.

7. But we can take many numbers by increasing the integers using pop() function. And pushing those to the push() function we can calculate the value.

8. During execution of the program we have to take some (in these case that is three) integers gradually as input and then by entering an operator ( + , - , \* , / ) we will find our results.

**Output of the program**

****

**Program written in C language**

#include <stdio.h>

#include <stdlib.h>

#define MAX 100

int \*p; /\* will point to a region of free memory \*/

int \*tos; /\* points to top of stack \*/

int \*bos; /\* points to bottom of stack \*/

void push(int i)

{

if(p > bos){

printf("Stack Full\n");

return;

}

\*p = i;

p++;

}

int pop(void)

{

p--;

if(p < tos) {

printf("Stack Underflow\n");

return 0;

}

return \*p;

}

int main(void)

{

float a, b,c;

char s[80];

p = (int \*) malloc(MAX\*sizeof(int)); /\* get stack memory \*/

if(!p) {

printf("Allocation Failure\n");

exit(1);

}

tos = p;

bos = p + MAX-1;

printf("Four Basic Function (Summation,Subtract,Multiply,Divide) of Calculator\n");

printf("Enter 'q' to quit\n");

printf("\nEnter three integers to one by one and then enter operator to calculate\n");

do {

printf(": ");

gets(s);

switch(\*s) {

case '+':

a = pop();

b = pop();

c = pop();

push(a+b+c);

printf("\n\n(%.2f+%.2f+%.2f) = %.2f\n\n",c,b,a,(a+b+c));

break;

case '-':

a = pop();

b = pop();

c = pop();

push(a-b-c);

printf("\n\n(%.2f-%.2f-%.2f) = %.2f\n\n",c,b,a,c-b-a);

break;

case '\*':

a = pop();

b = pop();

c = pop();

push(b\*a\*c);

printf("\n\n(%.2f\*%.2f\*%.2f) = %.2f\n\n",c,b,a,b\*a\*c);

break;

case '/':

a = pop();

b = pop();

c = pop();

if(a==0) {

printf("Divide by 0\n\n");

break;

}

push(c/b/a);

printf("\n\n(%.2f/%.2f/%.2f) = %.2f\n\n",c,b,a,c/b/a);

break;

case '.': /\*show contents of top of stack\*/

a = pop();

push(a);

printf("Current value on top of stack: %f\n", a);

break;

default:

push(atoi(s));

}

} while(\*s != 'q');

return 0;

}