**stack-using-array.c**

//write a program to implement stack using array.

#include <stdio.h>

#define max 10

int stack[10], top = -1;

void push(int data){

stack[++top] = data;

}

int pop(){

return stack[top--];

}

void display(){

int i = 0;

while(i <= top){

printf("%d",stack[i]);

i++;

if(i > top){

break;

}

printf(" -> ");

}

printf("\n");

}

int isEmpty(){

return top == -1;

}

int isFull(){

return top == max-1;

}

int len(){

return top+1;

}

void main(){

if(isEmpty()){

printf("The stack is empty.\n");

}else{

printf("The stack is not empty.\n");

}

printf("the length of the stack is %d\n",len());

push(10);

printf("the length of the stack is %d\n",len());

push(102);

push(15);

push(13);

push(12);

display();

printf("the length of the stack is %d\n",len());

if(isFull()){

printf("The stack is Full.\n");

}else{

printf("The stack is not Full.\n");

}

printf("Removed %d\n",pop());

printf("Removed %d\n",pop());

printf("Removed %d\n",pop());

display();

}

**OUTPUT**

PS S:\WorkSpace\CollegeWork\DataStructure\Temp> gcc .\stack-using-array.c

PS S:\WorkSpace\CollegeWork\DataStructure\Temp> ./a

The stack is empty.

the length of the stack is 0

the length of the stack is 1

10 -> 102 -> 15 -> 13 -> 12

the length of the stack is 5

The stack is not Full.

Removed 12

Removed 13

Removed 15

10 -> 102

PS S:\WorkSpace\CollegeWork\DataStructure\Temp>