Bahria University,

Karachi Campus

A picture containing text, room

Description automatically generated

LAB EXPERIMENT NO.

**05**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
|  | **STACK** |
| 1 | Write a program to build your own stack class. The minimum your stack class should include is using your enrollment no :   * + - A Push(Object) method     - A Pop() method     - A Peek() method     - A IsFull() method     - A IsEmpty() method     - A Display() method     - A Count() method |

Submitted On:

(Date: DD/MM/YY)

TASK NO # 1: Write a program to build your own stack class. The minimum your stack class should include is using your enrollment no :

* + - A Push(Object) method
    - A Pop() method
    - A Peek() method
    - A IsFull() method
    - A IsEmpty() method
    - A Display() method

A Count() method

**SOLUTION:**

class Program

{

internal class STACK

{

int top;

static readonly int max = 1000;

int[] stacck = new int[max];

public bool isempty() {

return (top < 0);

}

public STACK() {

top = - 1;

}

internal bool push(int data) {

if (top>=max)

{

Console.WriteLine("stack over flow");

return false;

}

else

{

stacck[++top]=data;

return true;

}

}

internal bool isfull() {

if (top==max)

{

Console.WriteLine("Stack is full");

return true;

}

else

{

Console.WriteLine("Stack is under Flow");

return true;

}

}

internal void printstack() {

if (top<0)

{

Console.WriteLine("Stack is underflow");

return;

}

else

{

Console.WriteLine("Elements of Stack");

for (int i = top; i >=0 ; i--)

{

Console.WriteLine(stacck[i]+" ");

}

}

}

internal int pop()

{

if (top<0)

{

Console.WriteLine("underflow :");

return 0;

}

else

{

int value = stacck[top--];

return value;

}

}

internal void peek() {

if (top<0)

{

Console.WriteLine("underflow :");

}

else

{

Console.WriteLine("The Peek Value Of Stack Is: {0}",stacck[top]);

}

}

internal int count() {

int con = 0;

if (top<0)

{

Console.WriteLine("underflow :");

return 0;

}

else

{

for (int i = top; i >=0; i--)

{

con++;

}

Console.WriteLine("total element In Stack Are :{0}",stacck[con]);

}

return con;

}

}

static void Main(string[] args)

{

STACK mystack = new STACK();

mystack.push(1);

mystack.push(2);

mystack.push(3);

mystack.push(4);

char ch;

do

{

Console.WriteLine("WHAT FUNCTION YOU WANT TO PERFORM :");

Console.WriteLine("1)push\n2)pop\n3)peek\n4)IsFull\n5)IsEmpty\n6)Display\n7)Count");

int option = int.Parse(Console.ReadLine());

if (option==1)

{

mystack.printstack();

}

else if (option == 2)

{

mystack.pop();

mystack.printstack();

}

else if (option == 3)

{

mystack.peek();

}

else if (option == 4)

{

mystack.isfull();

}

else if (option == 5)

{

Console.WriteLine("Your Stack Is Empty:" + mystack.isempty());

}

else if (option == 6)

{

mystack.printstack();

}

else if (option == 7)

{

mystack.count();

}

else

{

Console.WriteLine("Invalid");

}

Console.WriteLine("If you want to perform again [y/n]");

ch = char.Parse(Console.ReadLine());

Console.WriteLine("==================================================");

} while (ch=='y'||ch=='Y');

Console.ReadLine();

}

}}

**OUTPUT:**

