Bahria University,

Karachi Campus



COURSE: CSC-221 DATA STRUCTURES AND ALGORITHM

TERM: FALL 2020, CLASS: BSE- 3 (A)

Submitted By:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(ADIL WAHEED) (65190)

Submitted To:

Engr. Dr. Farah/ Engr. Ramshaa

Signed Remarks: Score:

INDEX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SNO | DATE | LAB NO | LAB OBJECTIVE | SIGN |
| 01 | 1-10-2020 | 01 | ONE AND TWO DIMENSIONAL ARRAY |  |
| 02 | 09-10-20 | 02 | Linear Search & Sorting Algorithms |  |
| 03 | 13-10-20 | 03 | Recusrion |  |
| 04 | 30/10/2020 | 04 | Binary Search Algorithm |  |
| 05 | 30/10/2020 | 05 | Merge Sort |  |
| 06 | 30/10/2020 | 06 | Quick Sort |  |
| 07 | 4/11/2020 | 07 | Stack |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| SNO | DATE | LAB NO | LAB OBJECTIVE | SIGN |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

\_\_\_07\_\_\_\_

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | 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: 4/11/20)

**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:**

internal class stack

{

int[] array = new int[20];

int top = -1;

internal void push(int x)

{

top = top + 1;

array[top]=x;

}

internal void pop()

{

if (top==-1)

{

Console.WriteLine("Please Enter Element is stack!!!!!!!");

}

else

{

Console.WriteLine("The Pop Element of stack is:{0}", array[top--]);

}

}

internal void peek()

{

Console.WriteLine("The Top Elememt of stack is:{0}", array[top]);

}

internal bool isempty()

{

if (top==-1)

{

Console.WriteLine("Stack is Empty");

return true;

}

Console.WriteLine("Stack is not Empty");

return false;

}

internal void count()

{

Console.WriteLine("stack Lenght={0}", top+1);

}

internal void isfull()

{

if (top==array.Length-1)

{

Console.WriteLine("Stack is Full");

}

Console.WriteLine("Stack is Not full");

}

internal void display()

{

if (top==-1)

{

Console.WriteLine("Stack is empty");

}

else

{

Console.WriteLine("Stack is:");

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

{

Console.WriteLine(“ “ + array[i] + " ");

}

Console.WriteLine();

}

}

}

static void Main(string[] args)

{

int a,ans;

char x;

stack s = new stack();

Console.WriteLine(" Enter Your Enrollment No in stack");

Console.WriteLine(" Push An Element=press 1");

Console.WriteLine(" Pop An Element Of Stack=press 2");

Console.WriteLine(" Peek Of An Element Of Stack=press 3");

Console.WriteLine(" Check Stack is Full=press 4");

Console.WriteLine(" Check Stack is Empty=press 5");

Console.WriteLine(" Display Of A Stack=press 6");

Console.WriteLine(" Count of Stack=press 7");

do

{

a = Convert.ToInt32(Console.ReadLine());

if (a == 1)

{

Console.WriteLine("Enter Your Enrollment No In Stack:");

for (int i = 0; i < 11; i++)

{

ans = Convert.ToInt32(Console.ReadLine());

s.push(ans);

}

}

else if (a==2)

{

s.pop();

}

else if (a == 3)

{

s.peek();

}

else if (a == 4)

{

s.isfull();

}

else if (a == 5)

{

s.isempty();

}

else if (a == 6)

{

s.display();

}

else if (a == 7)

{

s.count();

}

Console.WriteLine("Do you perform Anothr operation:y/n");

x = Convert.ToChar(Console.ReadLine());

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

**OUTPUT**:

