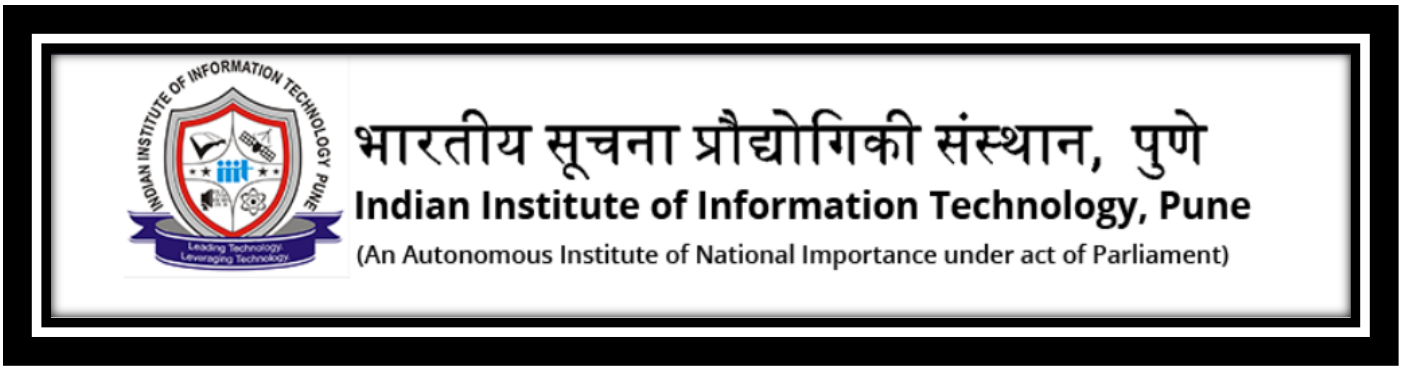


Data Structure and Algorithms

Semester II 2021-22

Lab- 3

Topics: Data Structure: Stacks

- GitHub Repo Link: Click Here

Exercise 3.a

SUDO CODE:

Create a .h file inside of it use class queue

Put all basic operation push , pop , display, top , is full , is empty

Using linked list (node class)

Take two pointers top and rear inside that class

Use void Push(int x) to push element

node temp = new node;*

temp->data=x;

if(rear==nullptr) {top=temp; rear=temp; }

else {rear->next=temp; rear=temp;

rear->next = nullptr;}

Use void Pop () to pop first element using rear pointer

*node *ptr ; ptr=top;*

if(ptr==top and ptr==rear){

top = nullptr ; rear = nullptr ; } else {while(ptr->next! =rear) {

ptr=ptr->next; } top = ptr;

*node *qtr = ptr->next ;*

top->next = nullptr ; delete qtr;} }

Inside main function create two objects of queue as global variable no change in push function but inside pop function use

Use a while loop so that it can point to the last element of the queue in this way it will implement stack

T*ime Complexity: O(n)*

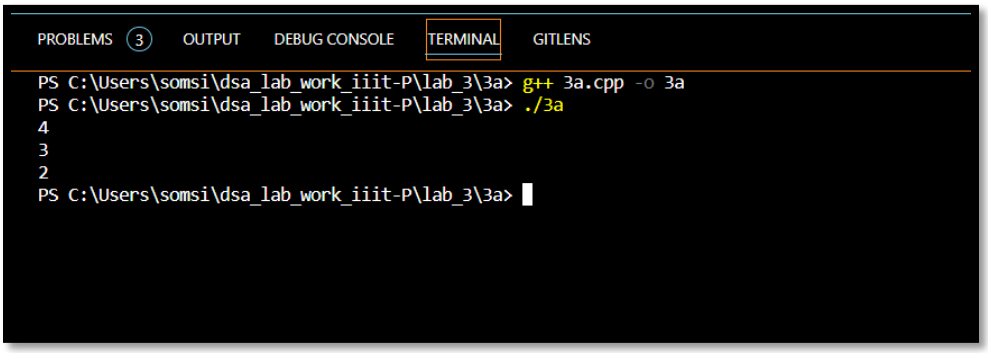
INPUT:

```
int main()
{
    push(1);
    push(2);
    push(3);
    push(4);

cout<< pop()<<endl;  cout<< pop()<<endl;  cout<< pop()<<endl;


                You, 10 hours ago • 3a.cpp works fine until we pop
    return 0;
}
```

OUTPUT:



Exercise 3.b

SUDO CODE:

Create a .h file inside of it use class stack

Put all basic operation push , pop , display, top , is full , is empty

Using linked list (node class) inside stack take top pointer which will point to the latest(last) element of the stack

Use void Push(int x) to push element

node temp = new node;*

temp->data=x;

if(rear==nullptr) {top=temp; rear=temp; }

else {rear->next=temp; rear=temp;

rear->next = nullptr;}

Use void pop () to delete the element

*Node *ptr = top*

Make top point to top->next and

Ptr->next = nullptr ;

Delete ptr;

Inside the main function make two stack object and declare them globally

int Pop () to implement queue

while(object1.size() > 1)

object2.push(object1.Top());

object1.pop();

while(object2.size()>0) object1.push(object2.Top());
object2.pop(); return temp;

Push() function will work same as it as in stack

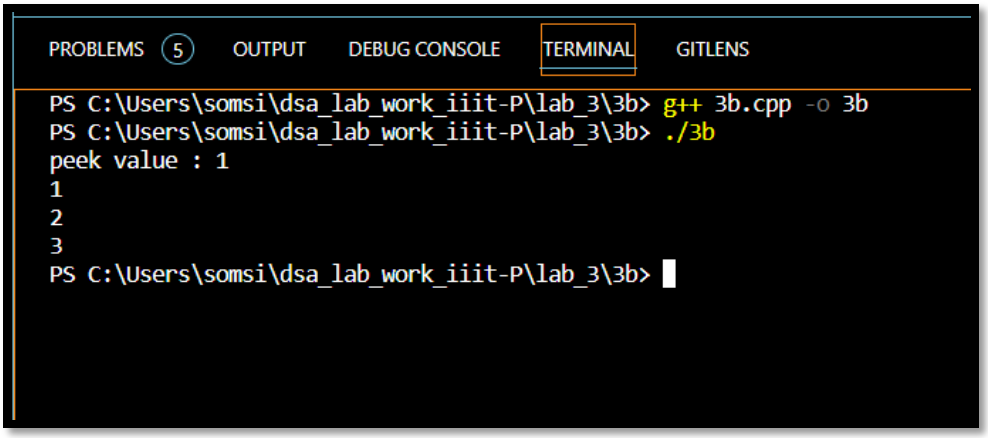
T*ime Complexity: O(n)*

INPUT:

```
int main()
{
    Push(1);
    Push(2);        You, 20 hours ago • initial commit
    Push(3);
    Push(4);
    Push(5);
    cout<<"peek value : " ;cout<<peek()<<endl;
    cout<<Pop()<<endl;
    cout<<Pop()<<endl;
cout<<Pop()<<endl;

    return 0;
}
```

OUTPUT:



Exercise 3.c

SUDO CODE:

Use same header that was used in stack ques(3b)

Take input of a string using cmd argument

Argv[1] = input string

Create stack A

Create bool matching parenthesis(string s) function which will take string as input

Initialize int c = 0 ;

Using a for loop when we counter '(' push that char

Else if '' increment c by 1*

else if(A.empty()) return false;

else if(A.Top()=='(' and s[i]!=')') return false;

Else pop the char

if(c==s.size()) if(c%2==0)return true; else return false;

else if(A.size()>0 && A.size()!=c)

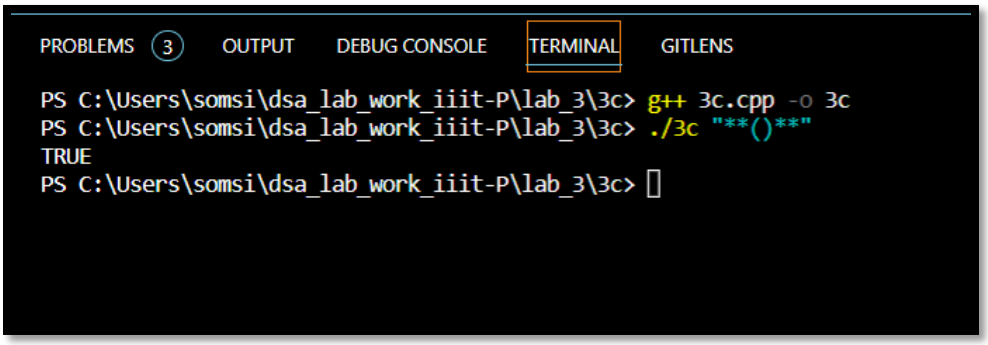
return false; else return true;

***T**ime Complexity: O(n)*

INPUT:

```
int main(int argc , char *argv[])
{       You, 10 hours ago • 3a.cpp works fine until
    string test_case = argv[1];
    // cout<<"Enter Your String"<<endl;
    // cin>>test_case;
    bool a=matching_parenthesis(test_case);
    if(a==true)
    {
        cout<<"TRUE"<<endl;
    }
    else{
        cout<<"FALSE"<<endl;
    }
    return 0;
```

OUTPUT:


PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
GITLENS
PS C:\Users\somsi\dsa_lab_work_iiit-P\lab_3\3c> g++ 3c.cpp -o 3c
PS C:\Users\somsi\dsa_lab_work_iiit-P\lab_3\3c> ./3c "**()**"
TRUE
PS C:\Users\somsi\dsa_lab_work_iiit-P\lab_3\3c>