**Department :** Computer Engineering

**Class :** SE

**Subject :** Data Structure Lab

**Name:** Kartiki Uday Khare.

**Roll no:** 21494

**Batch:** H4

**Assignment No : 09**

* **Problem Statement:**

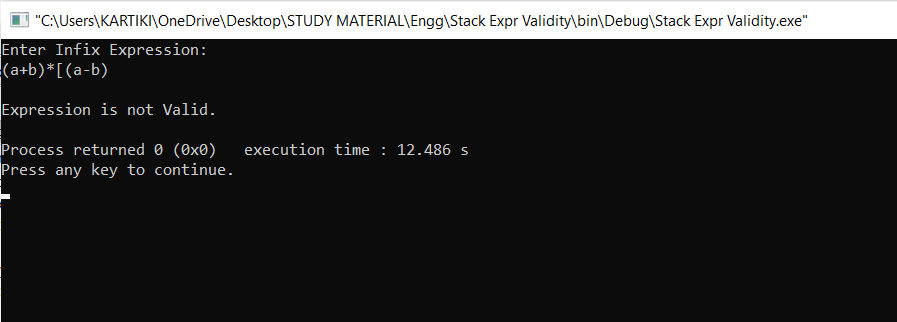
In any language program mostly syntax error occurs due to unbalancing delimiter such as (),{},[]. Write C++ program using stack to check whether given expression is well parenthesized or not.

* **Code:**

|  |
| --- |
| /\*  Name: Kartiki Uday Khare.  Roll No: 21494  Subject: DSL  Practical No: 09  \*/  #include <iostream>  #define max 100  using namespace std;  template<class T> //Declare Class Template  class Stack //Creating Stack  {  T arr[max];  int top;  public:  Stack();  void push(T val);  void pop();  T peep();  bool isempty();  void display();  };  template<class T>  Stack<T>::Stack()  {  top=-1;  }  template<class T>  void Stack<T>::push(T val)  {  if (top==max-1)  cout<<"\nStack is Full\n";  arr[++top]=val;  }  template<class T>  void Stack<T>::pop()  {  if(top!=-1)  top--;  }  template<class T>  T Stack<T>::peep()  {  if(top!=-1)  return arr[top];  }  template<class T>  bool Stack<T>::isempty()  {  if(top==-1)  return true;  else  return false;  }  template<class T>  void Stack<T>::display()  {  if (top!=-1)  {  cout<<"\nElements of Stack are:\n";  int x=top;  while(x!=-1)  {  cout<<arr[x]<<" ";  x--;  }  }  }  int main()  {  Stack<char>s;  string a;  cout<<"Enter Infix Expression:"<<endl;  cin>>a;  int i;  //bool ans=s.isempty();  for(i=0;i<a.length();i++)  {  if(s.isempty() && (a[i]=='(' || a[i]=='{' || a[i]=='[' || a[i]=='<'))  s.push(a[i]);  else if((s.peep()=='(' && a[i]==')') || (s.peep()=='{' && a[i]=='}') || (s.peep()=='[' && a[i]==']') || (s.peep()=='<' && a[i]=='>'))  s.pop();  else  continue;  }  if(s.isempty())  cout<<"\nExpression is Valid.\n";  else  cout<<"\nExpression is not Valid.\n";  return 0;  } |

* **Output:**

****

****