You are given a string of '(' and ')'. You have to check whether the sequence of parenthesis is balanced or not. For example, "(())", "(())()" are balanced and "())(", "(()))" are not.

Input Format

A string of '(' , ')' , '{' , '}' and '[' , ']' .

Constraints

1<=|S|<=10^5

Output Format

Print "Yes" if the parenthesis are balanced and "No" if not balanced.

Sample Input

(())

Sample Output

Yes

Program-

#include<iostream>

#include<bits/stdc++.h>

#include<stack>

#include<cstring>

using namespace std;

bool checkParenthesis(string str)

{

int n=str.length();

if(n%2==1)

return false;

stack<char> s;

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

{

if(str[0]=='}'||str[0]==']'||str[0]==')')

return false;

if(str[i]=='(' || str[i]=='{' || str[i]=='[')

s.push(str[i]);

if((str[i]==')') && (s.top()=='[' || s.top()=='{'))

return false;

if((str[i]==']') && (s.top()=='(' || s.top()=='{'))

return false;

if((str[i]=='}') && (s.top()=='(' || s.top()=='['))

return false;

if(str[i]=='}' && s.top()=='{' )

s.pop();

if(str[i]==']' && s.top()=='[')

s.pop();

if(str[i]==')' && s.top()=='(')

s.pop();

}

if(!s.empty())

return false;

else

return true;

}

int main() {

string str;

cin>>str;

if(checkParenthesis(str))

cout<<"Yes";

else

cout<<"No";

return 0;

}