**Assignment No.8(Paranthesized Expression).**

#include <iostream>

using namespace std;

class paranthesized{

char stacks[20],expr[20];

int top=-1;

public:

void accept(){

cout<<endl<<"Enter expression:";

cin>>expr;

}

void push(char ch){

stacks[++top]=ch;

}

char pop(){

return stacks[top--];

}

int match(char a,char b){

if(a=='{' && b=='}'){

return 1;

}else if(a=='(' && b==')'){

return 1;

}else if(a=='[' && b==']'){

return 1;

}else{

return 0;

}

}

int check(){

int i=0;

while(expr[i]!='\0'){

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

push(expr[i]);

}else if(expr[i]==')' || expr[i]==']' || expr[i]=='}'){

char temp=pop();

if(!match(temp,expr[i])){

return 0;

}

}

i++;

}

if(top!=-1){

return 0;

}else{

return 1;

}

}

};

int main()

{

paranthesized p1;

p1.accept();

int st=p1.check();

if(st==1){

cout<<endl<<"well";

}else{

cout<<endl<<"not";

}

return 0;

}**Output:**

Enter expression:{a+b]

This expression is not well paranthesized!!

Enter expression:{[{a+b}+(c-d)]}

This expression is well paranthesized!!