**LAB ASSIGNMENT 4**

**Ques1:** **Develop a menu driven program demonstrating the following operations on a Stack using array: (i) push(), (ii) pop(), (iii) isEmpty(), (iv) isFull(), (v) display(), and (vi) peek().**

#include<iostream>  
using  namespace std;  
int s[5];  
int top= -1;  
int value;  
  
void push(){  
  
  if(top==4){  
    cout<<"stack is full!";  
  }  
  else{  
    cout<<"enter stack elements: ";cin>>value;  
    s[++top]=value;  
    cout<<"value successfully pushed in the stack."<<endl;  
      }  
}  
  
void pop(){  
  if(top== -1){  
    cout<<"stack is empty!"<<endl;  
  }else{  
    cout<<s[top]<<" popped out of the stack."<<endl;  
    top--;  
  }  
}  
  
void display(){  
  if(top== -1){  
    cout<<"stack is empty!"<<endl;  
  }  
  else{  
    for(int i=top; i>=0;i--){  
      cout<<"elements in the stack are: "<< s[i]<<"\n";  
    }  
     
  }  
}  
  
int main(){  
  int choice;  
  while(true){  
   
  cout<<"\n--MENU--\n"<<endl;  
  cout<<"1.push"<<endl;  
  cout<<"2.pop"<<endl;  
  cout<<"3.display"<<endl;  
  cout<<"4.exit"<<endl;  
  cout<<"enter your choice: ";cin>>choice;  
   
  switch(choice){  
    case 1: push(); break;  
     case 2: pop(); break;  
     case 3: display(); break;  
     case 4: cout<<"exiting the proggramme.";  
     return 0;  
    default: cout<<"invalid choice.";  
  
  }  
}  
  
  return 0;  
}

**Ques2:** **Given a string, reverse it using STACK. For example “DataStructure” should be output as “erutcurtSataD.”**

#include <iostream>

#include <string>

using namespace std;

string reverseString(const string& s) {

string reversed = "";

for (int i = s.length() - 1; i >= 0; i--) {

reversed += s[i];

}

return reversed;

}

int main() {

string input;

cout<<"enter word you want to reverse:"; cin>>input;

cout << "Reversed String: " << reverseString(input) << endl;

return 0;

}

**Output:**

**enter word you want to reverse: StarWorld**

**Reversed String: dlroWratS**

**Ques3: Write a program that checks if an expression has balanced parentheses**

#include<iostream>

using namespace std;

int main() {

char str[100];

cout << "Enter an expression: ";

cin >> str;

char stack[100];

int top = -1;

int flag = 1;

int i = 0;

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

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

stack[++top] = str[i];

}

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

if (top == -1) {

flag = 0;

break;

}

char ch = stack[top--]; // pop

if ((str[i] == ')' && ch != '(') || (str[i] == '}' && ch != '{') ||

(str[i] == ']' && ch != '[')) {

flag = 0;

break;

}

}

i++;

}

if (top != -1)

flag = 0;

if (flag == 1)

cout << "Balanced" << endl;

else

cout << "Not Balanced" << endl;

return 0;

}

**Ques5:** **Write a program for the evaluation of a Postfix expression.**

#include <iostream>

#include <string>

using namespace std;

int findPostfix(const string& expr){

int stack[100];

int top=-1;

for(int i=0; i< expr.length();i++){

char ch= expr[i];

if(ch == ' ') continue;

if (isdigit(ch)){

int value= ch - '0';

stack[++top]= value;

}

else if(ch=='+' || ch=='-'||ch=='\*'||ch=='/'){

int a= stack[top--];

int b= stack[top--];

int result;

switch(ch){

case '+': result=a+b; break;

case '-': result=b-a; break;

case '\*': result=a\*b; break;

case '/': result=b/a; break;

}

stack[++top]=result;

}

}

return stack[top];

}

int main(){

string postfix;

cout<<"enter your postfix expression with space: ";

getline(cin, postfix);

int result = findPostfix(postfix);

cout<<"answer: "<< result;

return 0;

}

**Output:** **enter your postfix expression with space: 6 5 4 2 / + \***

**answer: 42**

**=== Code Execution Successful ===**