

**Name:** Lotfullah Muslimwal  
**Enrollment:** 01-131232-039

**Section:** BSE-3A  
**Course:** DSA-Lab  
**Lab No:** 03

**Task 1:** Create a Menu Base Program. That convert infix Expression to Postfix and infix to Prefix.

**Code:**

#include <iostream>

#include <stack>

#include <string>

using namespace std;

// it check that is entry an operand or not

bool Check\_Operand(char ch) {

return isalnum(ch); // isalnum is a build in function

}

// this function is used to check the precendence of the operator

int Check\_Precedence(char op) {

if (op == '+' || op == '-') {

return 1;

}

if (op == '\*' || op == '/') {

return 2;

}

if (op == '^') {

return 3;

}

// this part work as invalid input

return 0;

}

// this function is used to reverse the string for changing infix to prefix

string reverseString(string str) {

int n = str.length();

//loop for reversing the strings

for (int i = 0; i < n / 2; i++) {

swap(str[i], str[n - i - 1]);

}

return str;

}

//this function convert prefix to postfix

string toPostfix(string infix) {

stack<char> s;

string postfix = "";

for (char ch : infix) {

if (Check\_Operand(ch)) {

postfix += ch;

}

else if (ch == '(') {

s.push(ch);

}

else if (ch == ')') {

while (!s.empty() && s.top() != '(') {

postfix += s.top();

s.pop();

}

s.pop();

}

else {

while (!s.empty() && Check\_Precedence(s.top()) >= Check\_Precedence(ch)) {

postfix += s.top();

s.pop();

}

s.push(ch);

}

}

while (!s.empty()) {

postfix += s.top();

s.pop();

}

return postfix;

}

//this function convert infix to prefix

string toPrefix(string infix) {

infix = reverseString(infix);

for (char& ch : infix) {

if (ch == '(') ch = ')';

else if (ch == ')') ch = '(';

}

string prefix = toPostfix(infix);

prefix = reverseString(prefix);

return prefix;

}

int main() {

int choice;

string infix;

while (true) {

cout << "\nMenu:\n1. Convert Infix to Postfix\n2. Convert Infix to Prefix\n3. Exit\nEnter choice: ";

cin >> choice;

if (choice == 3) break;

cout << "Enter infix expression: ";

cin >> infix;

if (choice == 1) {

cout << "Postfix Expression: " << toPostfix(infix) << endl;

}

else if (choice == 2) {

cout << "Prefix Expression: " << toPrefix(infix) << endl;

}

else {

cout << "Invalid choice! Try again.\n";

}

}

return 0;

}

**GitHub-Link:** [**lmuslim2004/DSALAB3 (github.com)**](https://github.com/lmuslim2004/DSALAB3)

**Screenshot:**

