**Mechi Multiple Campus**

(Tribhuvan University)

Bhadrapur, Jhapa



**Lab Report of**

**Data Structures and Algorithm (CACS-201)**

**Conversion of Expression**

Faculty of Humanities & Social Sciences

Tribhuvan University

Kritipur, Nepal

**Submitted By**

**Name:** Santosh Bhandari

**Roll No:** 58

**Submitted To**

Mechi Multiple Campus

Department of Bachelor in Computer Application

Bhadrapur, Jhapa, Nepal

**Program Code**

#include<stdio.h>

#include<string.h>

#include<ctype.h>

#include<stdlib.h>

void push(char);

char pop(void);

int isoperator(char);

int isprecedence(char);

void InfixToPostfix(char[], char[]);

char stack[50];

int tos=-1;

void push(char data){

tos++;

stack[tos]=data;

}

char pop(){

char data;

data=stack[tos];

tos--;

return(data);

}

int isoperator(char symbol){

if(symbol == '^' || symbol == '$' || symbol == '\*' || symbol == '/' || symbol == '+' || symbol == '-')

return(1);

else

return(0);

}

int precedence(char symbol){

if(symbol == '^' || symbol == '$')

return(3);

if(symbol == '\*' || symbol == '/')

return(2);

if(symbol == '+' || symbol == '-')

return(1);

else

return(0);

}

void InfixToPostfix(char infix[], char postfix[]){

int i=0,j=0;

char scan\_el,x,op;

push('(');

strcat(infix,")");

scan\_el=infix[i];

while(scan\_el != '\0'){

if(isalpha(scan\_el)){

postfix[j]=scan\_el;

j++;

}else if(scan\_el == '('){

push(scan\_el);

}else if(isoperator(scan\_el)==1){

op=pop();

while(isoperator(op)==1 && precedence(op)>= precedence(scan\_el)){

postfix[j]=op;

j++;

op=pop();

}

push(op);

push(scan\_el);

}else if(scan\_el==')'){

x=pop();

while(x != '('){

postfix[j]=x;

j++;

x=pop();

}

}else{

printf("\nInvalid infix expression");

exit(0);

}

i++;

scan\_el=infix[i];

}

if(tos>0){

printf("\nInvalid infix expression");

exit(0);

}

postfix[j]='\0';

}

void main(){

char postfix[50], infix[50];

printf("\nInput the valid infix expression:");

gets(infix);

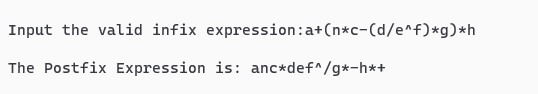
InfixToPostfix(infix,postfix);

printf("\nThe Postfix Expression is: ");

puts(postfix);

}

**Output of the Program**



**Program Code**

#include<stdio.h>

#include<math.h>

#include<string.h>

#include<ctype.h>

void push(int);

int pop();

int stack[50],tos=-1;

void main(){

char postfix[50],ch;

int i,a,b,c,value,result,len;

printf("Enter a Postfix Expression: ");

gets(postfix);

len=strlen(postfix);

for(i=0;i<len;i++){

ch=postfix[i];

if(isalpha(ch)){

printf("Enter the Value of %c: ",ch);

scanf("%d",&value);

push(value);

}

else{

a=pop();

b=pop();

switch(ch){

case '+':

c=b+a;

break;

case '-':

c=b-a;

break;

case '\*':

c=b\*a;

break;

case '/':

c=b/a;

break;

case '$':

case '^':

c=pow(b,a);

break;

}

push(c);

}

}

printf("Result = %d",pop());

}

int pop(){

int data;

data = stack[tos];

tos--;

return data;

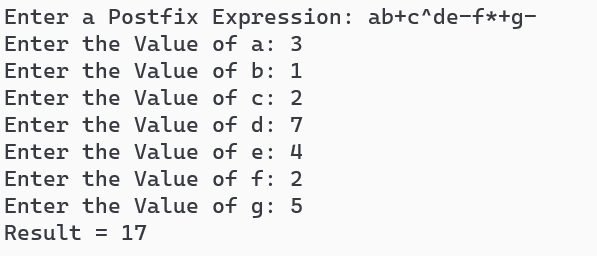
}

void push(int data){

tos++;

stack[tos]=data;

}

**Output of the Program**