



**SYSTEM
PROGRAMMING
LAB**

NEIL MATHEW – 6CS4 – Y3305 - A2324710002

Q 1: WAP to convert INFIX to POSTFIX notation.

[SOURCE CODE]

```
#include<stdio.h>
#include<conio.h>
#include<string.h>

#define MAX 20

char stack[MAX];
int top=-1;

char pop();
void push(char item);

int prcd(char symbol) // Precedence
{
    switch(symbol)
    {
        case '+':
        case '-':
            return 2;

        case '*':
        case '/':
            return 4;

        case '^':
            return 6;

        case '(':
        case ')':
        case '#':
            return 1;

        default:
            return 0;
    }
}

int isoperator(char symbol)
{
    switch(symbol)
    {
        case '+':    case '-':    case '*':    case '/':    case '^':
        case '(':    case ')':    return 1;

        default:    return 0;
    }
}
```

```

void convertip(char infix[],char postfix[])
{
    int i,symbol,j=0;
    stack[++top]='#';

    for(i=0;i<strlen(infix);i++)
    {
        symbol=infix[i];

        if(isoperator(symbol)==0)
        {
            postfix[j]=symbol;
            j++;
        }
        else
        {
            if(symbol=='(')
                push(symbol);

            else if(symbol==')')
            {
                while(stack[top]!='(')
                {
                    postfix[j]=pop();
                    j++;
                }
                pop();

            } else{

                if(prcd(symbol)>prcd(stack[top]))
                    push(symbol);

                else{

                    while(prcd(symbol)<=prcd(stack[top]))
                    {
                        postfix[j]=pop();
                        j++;
                    }

                    push(symbol);

                } //end of else.

            } //end of else.

        } //end of for.

        while(stack[top]!='#')
        {
            postfix[j]=pop();
            j++;
        }

        postfix[j]='\0'; //null terminate string.
    }
}

```

```
void main()
{
    char infix[20],postfix[20];

    clrscr();

    printf("Enter the valid infix string:\n");

    gets(infix);

    convertip(infix,postfix);

    printf("The corresponding postfix string is:\n");

    puts(postfix);

    getch();
}

void push(char item)
{

    top++;
    stack[top]=item;

}

char pop()
{

    char a;
    a=stack[top];
    top--;
    return a;

}
```

Q 1: WAP to convert INFIX to POSTFIX notation.

[SCREENSHOTS]

```
Enter the valid infix string:  
a+b/c  
The corresponding postfix string is:  
abc/+
```

```
Enter the valid infix string:  
a+b-(c^3)*(r/4)  
The corresponding postfix string is:  
ab+c3^r4/*-
```

Q 2: WAP to convert INFIX to PREFIX notation.

[SOURCE CODE]

```
#include<iostream.h>
#include<string.h>
#include<conio.h>
#include<ctype.h>

char stack[50];
int top=-1,x;
void push(char elem)
{
top++;
stack[top]=elem;
}
char pop()
{
x=stack[top];
top--;
return(x);
}
int order(char elem)
{
switch(elem)
{
case '#': return 0;
case '(': return 1;
case ')': return 1;
case '+':
case '-': return 2;
case '*':
case '/': return 3;
}

return 10; //default case
}

void main()
{
char infix[50],postfix[50],elem,ch;
int len,c=0,i=0;
cout<<"Enter the infix expresssion:";
cin>>infix;
push('#');
len=strlen(infix);
for(i=0;i<len;i++)
{
ch=infix[i];
if( ch == '(')
push(ch);
else if(isalpha(infix[i]))
{
postfix[c]=infix[i];
c++;
}
else if( ch == ')')
```

```

        {
            while(stack[top]!='(')
            {
                postfix[c]=pop();
                c++;
            }
            elem=pop();
        }
        else
        {
            while( order(stack[top]) >= order(ch) )
            {
                postfix[c]=pop();
                c++;
            }
            push(ch);
        }
    }
    while( stack[top] != '#')
    {
        postfix[c]=pop();
        c++;
    }
    postfix[c]='\0';
    cout<<"The Infix Expression is"<<" " <<infix<<endl;
    cout<<"The Postfix Expression is"<<" " <<postfix<<endl;
    getch();
}

```

Q 2: WAP to convert INFIX to PREFIX notation.

[SCREENSHOTS]

```
Enter the infix expresssion:a+b-c/d
The Infix Expression is a+b-c/d
The Postfix Expression is ab+cd/-
```

```
Enter the infix expression:a*b/r*c/(a-b)
The Infix Expression is a*b/r*c/(a-b)
The Postfix Expression is ab*r/c*ab-/-
```


Q 3: WAP to read and write a string to a file.

[SOURCE CODE]

```
//Using Turbo C

#include<fstream.h>
#include<conio.h>

#include<process.h> //for exit

#include<stdio.h> //for gets()

void main ()
{

clrscr();

int choice;

char data[100];
char cdata;

//using constructor
ofstream WriteFile("TheFile.txt",ios::out);
//using open() method
ifstream ReadFile;
ReadFile.open("TheFile.txt",ios::in);

while(1)
{
cout<<"\n\nEnter your choice"<<endl;
cout<<"Enter 1 to write to file"<<endl;
cout<<"Enter 2 to read string from file"<<endl;
cout<<"Enter 3 to read 5th character from file"<<endl;
cout<<"Enter 0 for exit"<<endl;
cout<<"\nEnter your choice: ";
cin>>choice;
cout<<endl;

switch(choice)
{
case 1:
cout << "Writing a string to the file" << endl;
cout << "Enter a string: ";
gets(data);

WriteFile.seekp(0, ios::beg);
WriteFile << data <<endl;

break;

case 2:
cout << "Reading a string from the file" << endl;
ReadFile.seekg(0, ios::beg);
ReadFile >> data;
cout << data << endl;
break;
```

```
case 3:
cout << "Reading 5th character from the file" << endl;
ReadFile.seekg(0, ios::beg);

for(int i=0; i<5; i++)
ReadFile >> cdata;

cout << cdata << endl;
break;

case 0:
ReadFile.close();
WriteFile.close();
exit(0);
}
}
}
```

Q 3: WAP to read and write a string to a file.

[SCREENSHOTS]

```
Enter your choice
Enter 1 to write to file
Enter 2 to read string from file
Enter 3 to read 5th character from file
Enter 0 for exit
```

```
Enter your choice: 1
```

```
Writing a string to the file
Enter a string: Hello...
```

```
Enter your choice
Enter 1 to write to file
Enter 2 to read string from file
Enter 3 to read 5th character from file
Enter 0 for exit
```

```
Enter your choice: 2
```

```
Reading a string from the file
Hello...
```

```
Enter your choice
Enter 1 to write to file
Enter 2 to read string from file
Enter 3 to read 5th character from file
Enter 0 for exit
```

```
Enter your choice: 3
```

```
Reading 5th character from the file
o
```

Q 4: WAP to perform lexical analysis of an expression.

[SOURCE CODE]

```
#include<iostream.h>
#include<string.h>
#include<ctype.h>
#include<conio.h>

void main()
{
    char s[50];
    char operands[50];
    char cons[50];
    char oper[50];
    int l=0,i=0,c=0,co=0,cou=0,count=0;
    cout<<"Enter a string";
    cin>>s;
    l=strlen(s);

    for(i=0;i<l;i++)
    {
        if(isalpha(s[i]))
        {
            operands[c]=s[i];
            c++;
        }
        else if(isdigit(s[i]))
        {
            cons[co]=s[i];
            co++;
        }
        else
        {
            oper[cou]=s[i];
            cou++;
        }
    }
    operands[c]='\0';
    cons[co]='\0';
    oper[cou]='\0';
    count=c+co+cou;
    cout<<"No of Operands:"<<" "<<c<<" "<<"\n\t They are:"<<"
"<<operands<<endl;
    cout<<"No of Operators:"<<" "<<cou<<" "<<"\n\t They are: "<<"
"<<oper<<endl;
    cout<<"No of Contants:"<<" "<<co<<" "<<"\n\tThey are: "<<"
"<<cons<<endl;
    cout<<"No of Tokens:"<<" "<<count<<endl;
    getch();
}
```

Q 4: WAP to perform lexical analysis of an expression.

[SCREENSHOTS]

```
Enter a string a*0=0
```

```
No of Operands: 1
```

```
They are: a
```

```
No of Operators: 2
```

```
They are: *=
```

```
No of Contants: 2
```

```
They are: 00
```

```
No of Tokens: 5
```

Q 5: WAP to accept a string of the form a(a+b)*abc.

[SOURCE CODE]

```
#include<iostream.h>
#include<string.h>
#include<ctype.h>
#include<conio.h>

void main()
{
    clrscr();
    char s[100];
    int l,flag=0,i=0;
    cout<<"Enter the string:"<<endl;
    cin>>s;
    l=strlen(s);
    if(s[0]=='a' && s[l-1]=='c' && s[l-2]=='b' && s[l-3]=='a')
    {
        for(i=1;i<l-3;i++)
        {
            if(s[i]!='a' && s[i]!='b' && s[i]!='+')
            {
                flag=1;
                cout<<"Format is not accepted"<<endl;
                break;
            }
        }
        if(flag==0)
            cout<<"Format is accepted"<<endl;
        else
            cout<<"Format is not accepted"<<endl;
    }
    else
        cout<<"Format is not accepted"<<endl;

    getch();
}
```

Q 5: WAP to accept a string of the form $a(a+b)^*abc$.

[SCREENSHOTS]

```
Enter the string:  
aababbbbabaabaabaaaaaaaaabc  
Format is accepted
```

```
Enter the string:  
aabc  
Format is accepted
```

```
Enter the string:  
bab  
Format is not accepted
```

Q 6: WAP to create a symbol table on analyzing the provided string.

[SOURCE CODE]

```
#include<iostream.h>
#include<stdio.h>
#include<string.h>
#include<ctype.h>

void main()
{
char in[50],dig[50],id[50];
int i=0,j=0,k,l=0;
cout<<"Enter the Expression:\t";
gets(in);
cout<<"\nKeyword\tIdentifier\tConstants\tOperators\tSpecialCharacters\n";

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

    if(isalpha(in[i]))
    {
        j=0;
        while((isalpha(in[i]))||(isdigit(in[i])))
        {
            id[j]=in[i];
            i++;
            j++;
        }
        id[j]='\0';

        if(strcmp(id,"char")==0||strcmp(id,"int")==0||strcmp(id,"float")==0||
strcmp(id,"if")==0 || strcmp(id,"then")==0 || strcmp(id,"while")==0 ||
strcmp(id,"do")==0||strcmp(id,"for")==0
        ||strcmp(id,"switch")==0||strcmp(id,"case")==0)
        {
            cout<<"\n";
            for(l=0;l<j;l++)
                cout<<" "<<id[l];

        }
        else
        {
            cout<<"\t";
            for(l=0;l<j;l++)
                cout<<id[l];

        }
    }
    else if(isdigit(in[i]))
    {
        k=0;
        while(isdigit(in[i]))
        {
            dig[k]=in[i];
            i++;
            k++;
        }
    }
}
```



```

}

cout<<"\n\t\t\t";
for(l=0;l<k;l++)
cout<<dig[l];
}
else if(in[i]=='+'||in[i]=='-'
'||in[i]=='*'||in[i]=='/'||in[i]=='<'||in[i]=='>'||in[i]=='=')
{
cout<<"\t\t\t\t\t ";<<in[i];
i++;
}
else
if(in[i]==','||in[i]==':'||in[i]=='.'||in[i]=='('||in[i]==')'||in[i]=='{'||in[i]=='}')
{
cout<<"\t\t\t\t\t\t\t ";<<in[i];
i++;
}
else
i++;
cout<<"\n";
}
}

```

Q 6: WAP to create a symbol table on analyzing the provided string.

[SCREENSHOTS]



Q 7: WAP to create a text editor.

[SOURCE CODE]

```
#include<stdlib.h>
#include<iostream.h>
#include<conio.h>
#include<fstream.h>
#include<string.h>

char data[100];
ofstream of;
ifstream iif;
int ch;
char fname[10];

void main()
{
clrscr();

while(ch!=5)
{

cout<<" -- ";
cout<<"\n\n MAIN MENU \n";

cout<<" 1.Write \n";
cout<<" 2.Read \n";
cout<<" 3.Copy \n";
cout<<" 4.Paste \n";
cout<<" 5.Exit \n";
cout<<" Your Choice:";

cin>>ch;

switch(ch)
{
case 1:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"WRITING TO THE FILE:";
of.open(fname);
cout<<"\nEnter DATA:"<<"\n";
cin>>data;
of<<data;
of.close();
break;

case 2:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"\nREADING FROM THE FILE:";
iif.open(fname);
while(!iif.eof())
{
iif>>data;
cout<<endl;
cout<<data;
```

```
}
iif.close();
break;

case 3:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"\nCOPYING FROM THE FILE:";
iif.open(fname);

while(!iif.eof())
{
iif>>data;
}
iif.close();
cout<<"\nDATA COPIED";
break;

case 4:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"PASTING TO THE FILE:";
of.open(fname);
of<<data;
of.close();
cout<<"\nDATA PASTED";
break;

default:
exit(0);
getch();
}
}
}
```

Q 7: WAP to create a text editor.

[SCREENSHOTS]

```
MAIN MENU
1.write
2.Read
3.Copy
4.Paste
5.Exit
Your Choice:1

FILENAME=Mine.txt
WRITING TO THE FILE:
Enter DATA:
Hello

--

MAIN MENU
1.write
2.Read
3.Copy
4.Paste
5.Exit
Your Choice:2

FILENAME=Mine.txt

READING FROM THE FILE:
Hello

--

MAIN MENU
1.write
2.Read
3.Copy
4.Paste
5.Exit
Your Choice:3

FILENAME=Mine.txt

DATA COPIED

--

MAIN MENU
1.write
2.Read
3.Copy
4.Paste
5.Exit
Your Choice:4

FILENAME=Mine2.txt

PASTING TO THE FILE:
DATA PASTED
```

Q 8: WAP to find the union and intersection of the two sets.

[SOURCE CODE]

```
#include<stdlib.h>
#include<iostream.h>
#include<conio.h>
#include<fstream.h>
#include<string.h>

char data[100];
ofstream of;
ifstream iif;
int ch;
char fname[10];

void main()
{
clrscr();

while(ch!=5)
{

cout<<" -- ";
cout<<"\n\n MAIN MENU \n";

cout<<" 1.Write \n";
cout<<" 2.Read \n";
cout<<" 3.Copy \n";
cout<<" 4.Paste \n";
cout<<" 5.Exit \n";
cout<<" Your Choice:";

cin>>ch;

switch(ch)
{
case 1:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"WRITING TO THE FILE:";
of.open(fname);
cout<<"\nEnter DATA:"<<"\n";
cin>>data;
of<<data;
of.close();
break;

case 2:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"\nREADING FROM THE FILE:";
iif.open(fname);
while(!iif.eof())
{
iif>>data;
cout<<endl;
cout<<data;
```

```
}
iif.close();
break;

case 3:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"\nCOPYING FROM THE FILE:";
iif.open(fname);

while(!iif.eof())
{
iif>>data;
}
iif.close();
cout<<"\nDATA COPIED";
break;

case 4:
cout<<"\nFILENAME=";
cin>>fname;
cout<<"PASTING TO THE FILE:";
of.open(fname);
of<<data;
of.close();
cout<<"\nDATA PASTESD";
break;

default:
exit(0);
getch();
}
}
}
```

Q 8: WAP to find the union and intersection of the two sets.

[SCREENSHOTS]

```
Enter the number of elements in first set:
5

Enter the elements:
1 2 3 4 5

Enter the number of elements in second set:
5

Enter the elements:
4 5 6 7 8

Menu:
1. Find the Union
2. Find the Intersection
3. Exit
Enter your choice:
1

Element of resultant set
      1      2      3      4      5      6      7      8

Menu:
1. Find the Union
2. Find the Intersection
3. Exit
Enter your choice:
2

Element of resultant set
      4      5
```