

EX 5 APPLICATION OF STACK

(Infix to post fix)

```
#include <stdio.h>
#include <string.h>
#define MAX 20
int Stack[MAX], top = -1;
char expr[MAX], post[MAX];
void Push(char sym);
char Pop();
char Top();
int Priority(char sym);
int main()
{
    int i;
    printf("Enter the infix expression : ");
    gets(expr);
    for(i = 0; i < strlen(expr); i++)
    {
        if(expr[i] >= 'a' && expr[i] <= 'z')
            printf("%c", expr[i]);
        else if(expr[i] == '(')
            Push(expr[i]);
        else if(expr[i] == ')')
        {
            while(Top() != '(')
                printf("%c", Pop());
            Pop();
        }
        else
        {
            while(Priority(expr[i]) <= Priority(Top()) && top != -1)
                printf("%c", Pop());
            Push(expr[i]);
        }
    }
    for(i = top; i >= 0; i--)
        printf("%c", Pop());
    return 0;
}

void Push(char sym)
{
    top = top + 1;
```

```

    Stack[top] = sym;
}
char Pop()
{
char e;
    e = Stack[top];
    top = top - 1;
return e;
}
char Top()
{
return Stack[top];;
}
int Priority(char sym)
{
int p = 0;
switch(sym)
{
case '(':
p = 0;
break;
case '+':
case '-':
p = 1;
break;
case '*':
case '/':
case '%':
p = 2;
break;
case '^':
p = 3;
break;
}
return p;
}

```

OUTPUT:

```

Enter the infix expression: ((a+b) * (c+d) * (e/f) ^g)

Postfix expression is =>
ab+cd+*ef/*g^aids231801049@csela

```