

Lab program:

```
#include <stdio.h>
#include <string.h>
#include <process.h>
int F (Char Symbol)
```

```
{
    Switch (Symbol)
```

```
    Case '+' :
```

```
    Case '-' : return 2;
```

```
    Case '*' :
```

```
    Case '/' : return 4;
```

```
    Case '^' :
```

```
    Case '$' : return 5;
```

```
    Case 'C' : return 0;
```

```
Case '?' : return -1;
```

```
default : return 8;
```

```
{
```

```
int G (Char Symbol)
{
    switch (Symbol)
    {
```

```
        Case '+':
```

```
        Case '-': return 1;
```

```
        Case '*':
```

```
        Case '/': return 3;
```

```
        Case '^':
```

```
        Case '$': return 6;
```

```
        Case 'C': return 9;
```

```
        Case 'e)': return 0;
```

default: return 7;

void infix-postfix (Char infix [], Char postfix [])

int top, i, j;

Char s[30], symbol;

top = -1;

s[top++] = '@';

j = 0;

for (i = 0; i < strlen(infix); i++)

{
symbol = infix[i];

while (FC(s[top]) > GC(symbol))

{

postfix[j] = s[top--];
j++;

}


```

if (FLS[top] != G(symbol))
{
    S[++top] = symbol;
}
else
{
    top top--;
}
while (S[top] != '#')
{
    postfix[j++] = S[top--];
}
postfix[j] = '\0';
}

void main()
{
    char infix[20];
    char postfix[20];
    clrscr();
    printf("Enter the valid infix exp\n");
    scanf("%s", infix);
    infix-postfix(infix, postfix);
    printf("the postfix exp is\n");
    printf("%s", postfix);
    getch();
}
    
```