

30-09-2020

Program - 2 :

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

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

```
#include <stdlib.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 '(': return 9;

case ')': 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 (P(s[top]) > Q(symbol))

{

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

j++;

}


```
if (F(s[top]) != G(symbol))
```

```
    s[++top] = symbol;
```

```
else
```

```
    top --;
```

```
}
```

```
while (s[top] != '#')
```

```
{
```

```
    postfin[j++] = s[top--];
```

```
}
```

```
    postfin[j] = '\0';
```

```
}
```

```
void main()
```

```
{
```

```
    char infix[20];
```

```
    char postfin[20];
```

```
    printf("Enter the valid infix expression\n");
```

```
    scanf("%s", infix);
```

```
    infix - postfin (infix, postfin);
```

```
    printf("The postfin expression is\n");
```

```
    printf("%s\n", postfin);
```

```
}
```