Infix to Post fix :—

```c
# include <stdio.h>
# include <stdlib.h>
# include <stdlib.h>
# include <string.h>

int F (char symbol) {
    switch (symbol) {
    case '+':
    case '-': return 2;
    case '*':
    case '/': return 4;
    case '^':
    case '$': return 5;
    case '(': 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;
```

```c
            case '(' : return 9;
            case ')' : return 0;
            case '#' : return -1;
            default : return 7;
        }
    }

void conversion (char infix[], char postfix[]) {
    int top, i, j;
    char s[50], symbol;
    top = -1;
    s[++top] = '#';
    j = 0;

    for (i=0; i < strlen(infix); i++) {
        symbol = infix[i];
        while (F(s[top]) > G(symbol))
        {

            postfix[j] = s[top--];
            j++;
        }
        if (F(s[top]) != G(symbol)) {
            s[++top] = symbol;
        }
        else {
            top--;
        }
    }

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

```c
        postfix[j++] = s[top--];
    }
    postfix[j] = '\0';
}
int main() {
    char infix[20];
    char postfix[20];
    printf("Enter expression: ");
    scanf("%s", infix);
    printf("
    conversion(infix, postfix);
    printf("The postfix expression : %s", postfix);
    return 0;
}
```