

15 . Write a C program to implement infix to postfix

```
#include <stdio.h>

#include <string.h>

char stack[100];

int top = -1;

void push(char c) {

    top++;

    stack[top] = c;

}

char pop() {

    char c = stack[top];

    top--;

    return c;

}

int priority(char c) {

    if (c == '*' || c == '/')

        return 2;

    else if (c == '+' || c == '-')

        return 1;

    else

        return 0;

}

void infixToPostfix(char infix[], char postfix[]) {

    int i, j = 0;

    char ch;

    for (i = 0; infix[i] != '\0'; i++) {
```

```

ch = infix[i];

if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a' && ch <= 'z')) {
    postfix[j] = ch;
    j++;
}

// If operator
else {
    while (top != -1 && priority(stack[top]) >= priority(ch)) {
        postfix[j] = pop();
        j++;
    }
    push(ch);
}

// Pop remaining operators
while (top != -1) {
    postfix[j] = pop();
    j++;
}

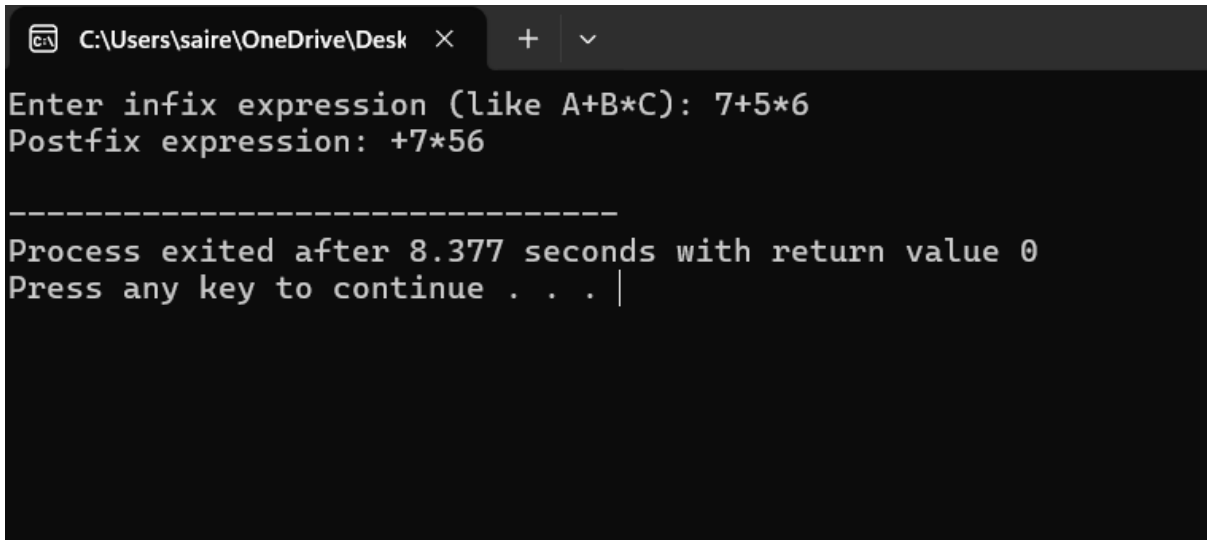
postfix[j] = '\0';
}

int main() {
    char infix[100], postfix[100];

```

```
printf("Enter infix expression (like A+B*C): ");  
  
scanf("%s", infix);  
  
infixToPostfix(infix, postfix);  
  
printf("Postfix expression: %s\n", postfix);  
  
return 0;  
}
```

Output



```
C:\Users\saire\OneDrive\Desk  ×  +  v  
Enter infix expression (like A+B*C): 7+5*6  
Postfix expression: +7*56  
  
-----  
Process exited after 8.377 seconds with return value 0  
Press any key to continue . . . |
```