

program 3: infix to postfix

Date _____	Page _____
------------	------------

```

code: void itp(char str[])
{
    for (int i=0; str[i]!='\0'; i++) {
        char ch = str[i];
        if ((ch>='A' && ch<='Z') || (ch>='a' && ch<='z')
            || (ch>='0' && ch<='9')) {
            printf("%c", ch);
        } else if (ch=='(') {
            push(ch);
        } else if (ch==')') {
            while (top != -1 && stack[top] != '(') {
                printf("%c", pop());
            }
            pop();
        } else {
            while (top != -1 && getPriority(ch) <= getPriority
                   (stack[top])) {
                printf("%c", pop());
                push(ch);
            }
        }
    }
}

```

3.7

Date: 14/10/25

Page: 31

Output: ab*c+d*ef

Enter an expression: a*b+c*d-e
ab*cd*ef-
ab*cd*ef-
ab*cd*ef-

OP: abcdef

writing of xini : 8 margin

code: #include <stdio.h> // main() g/f
#include <ctype.h> // for isdigit()

#define N('10')>>#A'-'0') fi
char stack[N]; // 0 = <int>
int top = -1; // (0, "0") f/nop
3(5=nd) fi 2/3

void push(char ch) { // (n) nanc
stack[++top] = ch; } // 2/3
3(4)= [got] 2/3 1 = (got) slider
char pop() { // (0, "0") f/nop
return stack[top--]; }
3

int getPriority(char op) { // (0) 2/3
switch (op) {

case '+' : return 1; // (1)
case '-' : return 1; // (1)
case '*' :
case '/' : return 2; // (2)
case '^' : return 3; // (3)
default : return 0; } // (0)

void itp(char str[]) { // (0)
for (int i = 0; str[i] != '\0'; i++) {
char ch = str[i];
if (ch >= 'A' && ch <= 'Z') || (ch >= 'a' && ch <= 'z') {
if (ch >= '0' && ch <= '9') {
printf("%c", -ch); } // 3/3
else if (ch == '(') {
push(ch); }
else if (ch == ')') {

```
while ( top != -1 && stack[top] != 'c' ) {  
    printf("%c", pop());  
    pop();  
}  
else {  
    while ( top != -1 && getPriority(ch) <= getPriority( stack[top] ) )  
        printf("%c", pop());  
    push(ch);  
}  
}  
while ( top != -1 ) {  
    if ( stack[top] != '(' )  
        printf("%c", pop());  
    else  
        pop();  
}  
}
```

```
int main () {  
    char = exp[50]  
    printf("Enter an expression: ");  
    scanf ("%[^/n]", exp);  
    itp(exp);  
    return 0;  
}
```

O P Sem