

```

1. //Implementation of Infix to Postfix Conversion
2. //save the program with anyname.c
3. #include<stdio.h>
4. #include<ctype.h>
5. char stack[100];
6. int top = -1;
7. void push(char x)
8. {
9.     stack[++top] = x;
10. }
11. char pop()
12. {
13.     if(top == -1)
14.         return -1;
15.     else
16.         return stack[top--];
17. }
18. int priority(char x)
19. {
20.     if(x == '(')
21.         return 0;
22.     if(x == '+' || x == '-')
23.         return 1;
24.     if(x == '*' || x == '/')
25.         return 2;
26.     return 0;
27. }
28. int main()
29. {
30.     char exp[100];
31.     char *e, x;
32.     printf("Enter the expression : ");
33.     scanf("%s",exp);
34.     printf("\n");
35.     e = exp;
36.
37.     while(*e != '\0')
38.     {
39.         if(isalnum(*e))
40.             printf("%c ",*e);
41.         else if(*e == '(')
42.             push(*e);
43.         else if(*e == ')')
44.         {
45.             while((x = pop()) != '(')
46.                 printf("%c ", x);
47.         }
48.         else
49.         {
50.             while(priority(stack[top]) >= priority(*e))
51.                 printf("%c ",pop());
52.             push(*e);
53.         }
54.         e++;
55.     }
56.
57.     while(top != -1)
58.     {
59.         printf("%c ",pop());
60.     }return 0;

```

61. }

62. Output:

63. yasar@yasar-Lenovo-G50-80:~/Desktop/DataStructures/lords/lab\$ gcc infixTopostfix.c -o ip2

64. yasar@yasar-Lenovo-G50-80:~/Desktop/DataStructures/lords/lab\$ ./ip2

65.

66. Enter the expression : A+B

67.

68. A B +

69.

70. yasar@yasar-Lenovo-G50-80:~/Desktop/DataStructures/lords/lab\$ ./ip2

71. Enter the expression : (A+B)\*(C+D)

72.

73. A B + C D + \*