

enter the valid infix expression
(A+(B-C)*D)
the postfix expression is

...Program finished with exit code 0
Press ENTER to exit console.[]

I

2. Map to convert a given valid parenthesized infix arithmetic expression to postfix expression. The expression consists of single character operands and the binary operators + (plus), - (minus), * (multiply) & / (divide).

```

=> #include <stdio.h>
#include <string.h>
int P (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 Q (char symbol) {
    switch (symbol)
    {

```

```

        case '+':

```

```

        case '-':

```

```

        case '*': return 1;

```

```

        case '/':

```

```

        case '^': return 7;
    }
}

```

```

case 'A' :
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 (F[s[top]] > G(symbol))
        {

```

```

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

```

```

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

```

```

            s[top++] = symbol;

```

```

        else
            top = -1;
        }
    }
}

```


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

{

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

}

postfix[j] = '\0';

}

Void main()

{

char infix[20];

char postfix[20];

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

scanf("%s", infix);

infix -> postfix(infix, postfix);

printf("The postfix expression is\n");

printf("%s\n", postfix);

}

Output:

Enter valid infix expression

(A + (B - C) * D)

the postfix expression is

AB(C-D)*+