```
Run >
main.c
                                                                                       1
      #include<stdio.h>
  2
      #include<string.h>
  3
      #include<stdlib.h>
  4
  5
      int F(char symbol)
  6
      {
  7
      switch(symbol)
  8
  9
      case'+':
 10
       case'-':return 2;
 11
       case'*':
 12
       case'/':return 4;
 13
       case'^':
 14
       case'$':return 5;
 15
       case'(':return 0;
       case'#':return -1;
 16
 17
       default:return 8;
 18
       }
 19
                                                                                     I
 20
 21
       int G(char symbol)
 22
       {
       switch(symbol)
 23
 24
        {
 25
       case'+':
 26
       case'-':return 1;
        case'*':
 27
 28
        case'/':return 3;
       case'^':
 29
       case'$':return 6;
 30
       case'(':return 9;
 31
        case')':return 0;
 32
 33
        default: return 7;
 34
       }
 35
       }
 36
 37
      void infix_postfix(char infix[],char postfix[])
 38
       {
 39
      int top, i, j;
      char s[30],symbol;
 40
 41
      top=-1;
 42
       s[++top]='#';
 43
      j=0;
 44
 45
       for(i=0;i<strlen(infix);i++)</pre>
```

```
et C v 🕤
                                                                      Run >
  main.c
                                                                                            42
         s[++top]='#';
    43
         j=0;
    44
         for(i=0;i<strlen(infix);i++)</pre>
    45
    46
          {
         symbol=infix[i];
    47
    48
         while(F(s[top])>G(symbol))
    49
          {
          postfix[j]=s[top--];
    50
    51
          j++;
    52
          }
    53
     54
     55
          if(F(s[top])!=G(symbol))
     56
          s[++top]=symbol;
     57
          else
                                                                    I
     58
          top--;
     59
          }
     60
     61
          while(s[top]!='#')
     62
           {
     63
          postfix[j++]=s[top--];
     64
     65
           postfix[j]='\0';
     66
      67
      68
           int main()
      69
      70
           char infix[20];
      71
           char postfix[20];
           printf("Enter the valid infix expression ");
      72
      73
           scanf("%s", infix);
           infix_postfix(infix , postfix );
      74
           printf("The postfix expression is \n");
      75
      76
           printf("%s\n", postfix);
      77
```

```
clang-7 -pthread -lm -o main main.c

./main

Enter the valid infix expression (1+2)*(3/4)*(6-2)

The postfix expression is

12+34/*62-*

. []
```

S.R.POOJA 1BM19CS135 CSE DS LAB	classmate  Date Page
#include < stdio.h>	2 -
# include ( steing . h >	i i och di
# welude < stdlib.h >	: Jodnys (1872 sept)
int F (char symbol)	1 007
£	"#" .[ DIT+12
Switch ( Symbol)	0 5 %
3	171 (1010) olds > 1:0=1 ) Jal
case '+'!	3
case '-': setuen 2;	Gil rate = labouth
case '*':	(Clodenie ) DC (Cool 7 e 13) elister
case '/': setuen 4;	7
case 'N':	[gat]e =[i] myan
case '\$': setuen 5',	***
case 'C' : setuen 0;	
	((Johns 2) 0= 1(Top) 2) ji
default: seturn 8;	Jahrys of Oct + 78
3	ele
3	· 00
int Gr (char symbol)	('-4+1 = 1 (got) 8) alistra
Switch (Symbol)	3
£	Control of the - ];
case '+':	
(age'-'; Return 1;	i'or' of all a idea
cak '*':	
case 1/ = gettin 3;	Quen ta
case 'A':	3
case '\$' = getter 6;	i Cachnia acts
case (' : getuen 9;	
(ase ')' \$: geturn 0;	
default: letuer 7;	
3	Costing which after which
3 " (")	Lagre ( " The mother commercian
void infix postfix (cho	or infix [], char postfix [])

```
int top, ij;
chas & [30], symbol;
top = -1;
&[++top]= '#'
for ( i=0; i < stelen (infor); i++)
Symbol = infor [i];
while (F(8[top])>G(8ymbol))
postax [j]= s[top--];
if (F(s [top])! = Gr (symbol))
S[++top]= symbol;
while (8[top] ! = \#')
postfore[j++]= &[top--];
 postfix (j)= 6 101;
int main ()
char infor (20);
chae postfix [20],
prints ("Enter the valid infox ex expression");
scarf (" / & infox);
people" The postfox expression is In").

print[ " 1.8 \n", postfox);
```