

Date: 05/10/2000 USN: 18M19CS137 Mame: 55kander

WAP to convert a given valid parenthesized infix arithmetic expression to postfix expression.

The expression consists of single character operands and the binary operators.

#include < string sh>
#include < string sh>
#include < process.h

int F (chear symbol)

{ switch (symbol)

{ case '+'

case '-': return 2;

case 'A':

case 'A':

case 'A': return 5;

case 'H': return 0;

case 'H': return 0;

case 'H': return 0;

case 'H': return 0;

default: return 8;

}

int G (char symbol)

E switch (symbol)

E case '+':

case '-': return 1;

case '/': return 3;

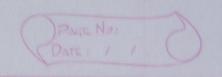
case '/': return 5;

```
case (: return 9;
       (ase) return oj
       default: return 7;
void infix-postfix (char infix(), char postfix())
   int i, i, top;
   char st30], symbol;
    st++top]= '#+';
    for (i=o; i < strlen (infix); i++)
      symbol = infix[i];
        while (F(s[top]) > G(symbol))

{ postfix (j]!= G(symbol)

postfix (j]=!s[top--];
         if (F(s[top]) != G(symbol))
s[++top]=symbol;
      while (s[top]!="#")

2 postfix [j++]=s[top--];
       postfix[j]='10';
```



E char infix[20], postfix[20];

printf ("Enter a valid infix expression;");

scapf ("7.5", & infix);

infix - postfix (infix, postfix);

printf ("The postfix expression is :");

printf ("7.5", postfix);

return 0;