```c
#include <stdio.h>
#include <string.h>
#include <conio.h>
int F (char symbol)
{
    switch (symbol)
    {
    case '+':
    case '-': return 1;
    case '*':
    case '/': return 2;
    case '$': return 6;
    case '(': return 0;
    case '#': return -1;
    default: return 8;
    }
}
int G(char symbol)
{
    switch (symbol)
    {
    case '+':
    case '-': return 2;
    case '*':
    case '/': return 4;
    case '$': return 5;
    case '(': return 0;
    case ')': return 9;
    default: return 7;
    }
}
```

```c
void infix_prefix (char infix[], char prefix[])
{
    int top, i, j;
    top = -1;
    char symbol, s[30];
    s[++top] = '#';
    j = 0;
    strrev(infix);
    for(i = 0; i < strlen(infix); i++)
    {
        symbol = infix[i];
        while (F[s[top]] > G(symbol))
        {
            prefix[j] = s[top--];
            j++;
        }
        if (F[s[top]] != G(symbol))
        {
            s[++top] = symbol;
        }
        else
        {
            top--;
        }
    }
    while (s[top] != '#')
    {
        prefix[j++] = s[top--];
    }
    prefix[j] = '\b';
    strrev(prefix);
}

void main()
{
    char infix[30];
    char prefix[30];
```

```
clrscr;
pf (" Enter valid infix exp: \n");
sf (" °l·s" , infix);
infix_prefix ( infix, prefix);
pf (" the prefix exp is ");
pf (" ·l·s\n" , prefix);
getch();
3.
```