```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>

#define SIZE 50


char stack[SIZE];
int top=-1;


void push(char elem)
{
   top++;
   stack[top]=elem;
   return ;
}

char pop()
{
   char x;
   x=stack[top];
   top--;
   return(x);
}

char peek()
{
   char x;
   x=stack[top];
   return(x) ;
```

```
}


int pr(char elem)
{
   switch(elem)
   {
   case '#': return 0;
   case '(':  return 1;
   case '+':
   case '-':  return 2;
   case '*':
   case '/':  return 3;
   }
}

void main()
{
   char infx[50],pofx[50],ch,elem;
   int i=0,k=0;
   clrscr();

   printf("\n\nRead the Infix Expression ? ");
   scanf("%s",infx);
   push('#');

   while( (ch=infx[i++]) != '\0')
   {
     if( ch == '(') push(ch);
     else
     if(isalnum(ch)) pofx[k++]=ch;
```

```
    else
    if( ch == ')')
     {
          while( peek() != '(' )
         pofx[k++]=pop();
          elem=pop();
     }
   else
     {
          while( pr(peek()) >= pr(ch) )
         pofx[k++]=pop();
          push(ch);
     }
   }

  while( peek() != '#')
      pofx[k++]=pop();

  pofx[k]='\0';
  printf("\n\nGiven Infix Expn: %s  Postfix Expn:
%s\n",infx,pofx);
  getch();
}
```