

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

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

```
#define SIZE 50
```

```
char stack[SIZE];
```

```
int top=-1;
```

```
void push(char elem)
```

```
{
```

```
    stack[++top]=elem;
```

```
}
```

```
char pop()
```

```
{
```

```
    return stack[top--];
```

```
}
```

```
int pr(char symbol)
```

```
{
```

```
    if(symbol=='^')
```

```
    {
```

```
        return(3);
```

```
    }
```

```
    else if(symbol=='*' | symbol=='/')
```

```
    {
```

```
        return(2);
```

```
    }
```

```
    else if(symbol=='+' | symbol=='-')
```

```

    {
        return(1);
    }
else
    {
        return(0);
    }
}

```

```

int main()
{
    char infix[50],postfix[50],ch,elem;
    int i=0,k=0;

    printf("Enter the Infix expression: ");
    scanf("%s",&infix);

    push('#');

    while((ch=infix[i++])!='\0')
    {
        if(ch=='(')push(ch);
        else
            if(isalnum(ch))postfix[k++]=ch;
        else
            if(ch==')')
            {
                while(stack[top]!='(')
                    postfix[k++]=pop();
            }
        }
    }
}

```

```

        elem=pop();
    }
    else
    {
        while(pr(stack[top])>=pr(ch))
            postfix[k++]=pop();
        push(ch);
    }
}

while(stack[top]!='#')
    postfix[k++]=pop();

postfix[k]='\0';

printf("\nPostfix Expression = %s\n",postfix);

return 0;

```

```
Enter the Infix expression: ((a*b)+c*d-e)
```

```
Postfix Expression = ab*cd*+e-
```

```
-----
(program exited with code: 0)
```

```
Press any key to continue . . .
```

```
}
```