

# Программирование задач с использованием функций

Практическое занятие №6

# Указатели и ссылки

Указатель – это переменная, которая содержит адрес другой переменной

Описание указателя:

<тип> \*<идентификатор>

Пример: int \*p;

p



p

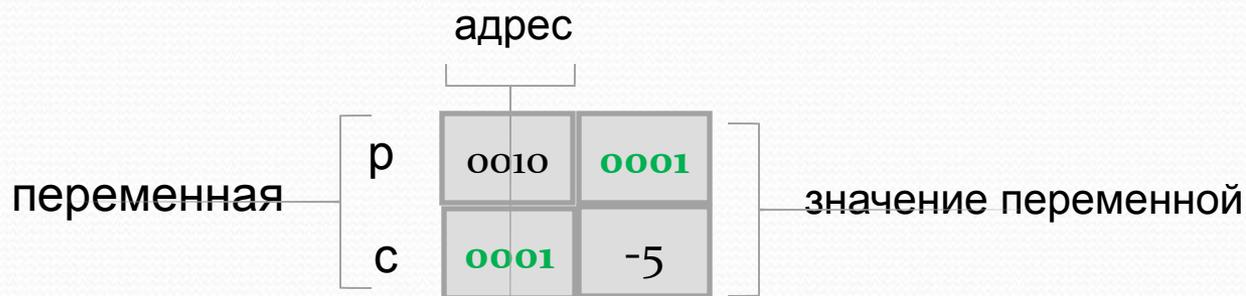


```
int *p;  
p=new int;  
*p=-5;
```

ИЛИ

```
int *p, c=-5;  
p=&c;
```

Оператор & -  
взятие адреса



# Функции в C++

При повторении **одних и тех же действий** в разных частях алгоритма при **различных значениях данных** необходимо выделять описания этих действий в отдельную **функцию** (подпрограмму)

#Директивы препроцессору  
Глобальные данные (переменные, константы)

Функция 1(список формальных параметров)  
Локальные данные

Функция 2(список формальных параметров)  
Локальные данные

Функция n(список формальных параметров)  
Локальные данные

Основная функция main()  
Локальные данные  
Функция1(фактические параметры)  
Функция2(фактические параметры)

# Общий вид функции:

```
<тип возвращаемого значения> <имя функции> (<формальный  
параметр 1>, <формальный параметр 2>, ...)  
{  
    тело функции  
    return <возвращаемое значение>;  
}
```

# Вызов функции:

```
<переменная>=<имя функции>(список фактических параметров);
```

# Примеры функций

```
int IsSign (double a)
{
    if(a>0) return 1;
    else    if(a==0) return 0;
           else return -1;
}
```

*1. параметр-константа.*

```
int main()
{
    int n,m;
    cout<<"n=";
    cin>>n;
    m=IsSign(n);
    switch(m)
    {
        case -1:cout<<"otritsat"<<endl;break;
        case 0:cout<<"NULL"<<endl;break;
        case 1:cout<<"polozhitel"<<endl;break;
    }
}
```

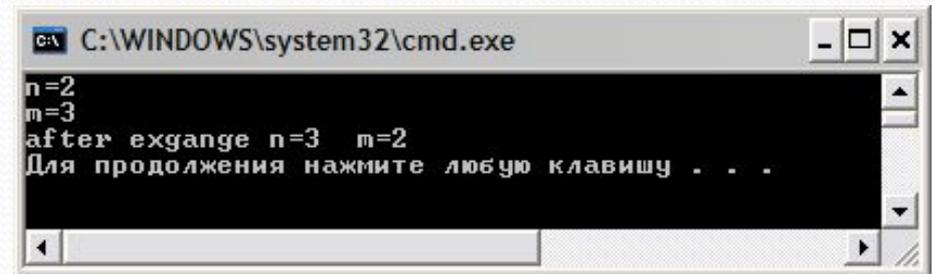
```
void exgange (int &a,int &b)
```

```
{  
    int s=a;  
    a=b;  
    b=s;  
}
```

```
int main()
```

```
{  
    int n,m;  
    cout<<"n=";  
    cin>>n;  
    cout<<"m=";  
    cin>>m;  
    exgange(n,m);  
    cout<<"after exgange n="<<n<<" m="<<m<<endl;  
}
```

## 2. параметр-переменная.



```
C:\WINDOWS\system32\cmd.exe  
n=2  
m=3  
after exgange n=3 m=2  
Для продолжения нажмите любую клавишу . . .
```

# Пример функции формирования и вывода массива

```
#include <iostream>
#include <iomanip>
using namespace std;
const int n=6;

void input(int a[])
{
    for(int i=0;i<n;i++)a[i]=rand()/100-100;
}

void output (int a[])
{
    cout<<"Output matr"<<endl;
    for(int i=0;i<n;i++)
        cout<<setw(6)<<a[i];
    cout<<endl;
}
```

# Основная программа

```
int main()
{
    int f[n];
    input(f);
    output(f);
}
```



A screenshot of a Windows command prompt window. The title bar shows the path "C:\WINXP\system32\cmd.exe". The window content displays the output of a program:

```
Output matr>
-100  84  -37  165  91  57
Для продолжения нажмите любую клавишу . . .
```

The window includes standard Windows window controls (minimize, maximize, close) and a scroll bar at the bottom.

# Другой заголовок функции

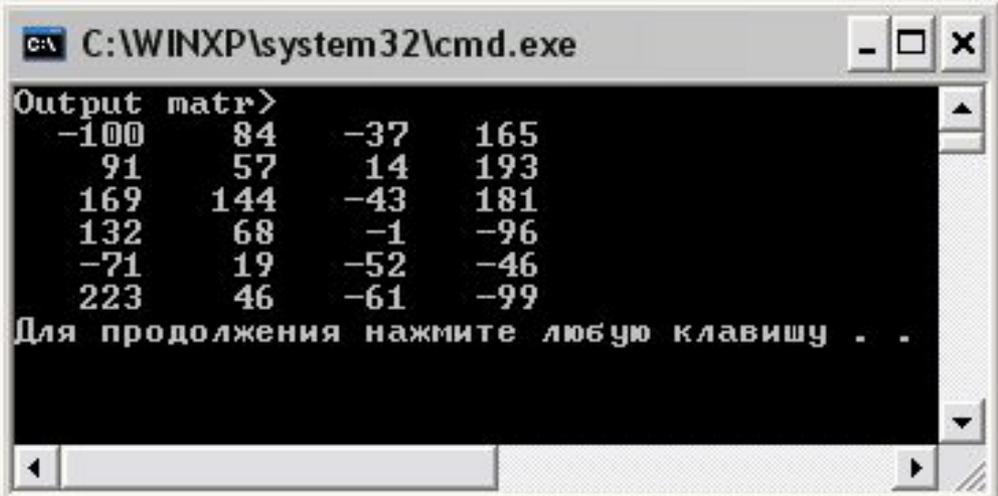
```
void input(int *a)
{
    for(int i=0;i<n;i++)a[i]=rand()/100-100;
}
void output (int *a)
{
    cout<<"Output matr"<<endl;
    for(int i=0;i<n;i++)
        cout<<setw(6)<<a[i];
    cout<<endl;
}
```

# Функции формирования и вывода матрицы

```
const int n=6, m=4;
void input( int a[][m])
{
    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            a[i][j]=rand()/100-100;
}
void output ( int a[][m])
{
    cout<<"Output matr"<<endl;
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<m;j++)
            cout<<setw(6)<<a[i][j];
        cout<<endl;
    }
}
```

# Основная программа

```
int main()
{
    int f[n][m];
    input(f);
    output(f);
}
```



```
C:\WINXP\system32\cmd.exe
Output matr>
-100    84    -37    165
  91    57     14    193
 169   144   -43    181
 132    68     -1   -96
 -71    19   -52   -46
 223    46   -61   -99
Для продолжения нажмите любую клавишу . .
```

# программирования

Даны две матрицы A и C. Сформировать одномерные массивы P и Q из количества положительных элементов каждого столбца соответствующих матриц.

```
#include <iostream>
#include <iomanip>
using namespace std;
typedef int matr[5][5];
typedef int mas[5];
void input(matr a, int n, int m)
{
    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            a[i][j]=rand()/100-100;
}
```

```
void output ( matr a, int n, int m)
{
    cout<<"Output matr"<<endl;
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<m;j++)
            cout<<setw(6)<<a[i][j];
        cout<<endl;
    }
}
```

```
void form(matr a,mas b,int n, int m)
{
    int kol;
    for(int j=0;j<m;j++)
    {
        kol=0;
        for(int i=0;i<n;i++)
            if(a[i][j]>0)kol++;
        b[j]=kol;
    }
}
```

```

void output_mas(mas a, int m)
{
    cout<<"massiv"<<endl;
    for(int i=0;i<m;i++)
        cout<<setw(6)<<a[i];
    cout<<endl;
}
int main()
{
    matr f; mas k;
    input(f,5,5);
    output(f,5,5);
    form(f,k,5,5);
    output_mas(k,5);
}

```

```

C:\WINDOWS\system32\cmd.exe
Output matr>
-100    84    -37    165    91
  57    14    193    169    144
 -43    181    132    68    -1
 -96   -71    19    -52   -46
 223    46   -61   -99   -98
massiv
  2     4     3     3     2
Для продолжения нажмите любую кла

```

Даны массивы A{7}, B{5}, C{9} и D{9}. Вычислить

$$p = \frac{\prod_{i=0}^7 a_i + \ln \prod_{j=0}^5 b_j}{\sin \prod_{k=0}^9 c_k * \sqrt{\prod_{n=0}^9 d_n}}$$

```
#include <iostream>
#include <iomanip>
#include <math.h>
using namespace std;

void input(double a[], int n)
{
    for(int i=0;i<n;i++) a[i]=i+1;
}
void output ( double a[], int n)
{
    cout<<"Output mas"<<endl;
    for(int i=0;i<n;i++) cout<<setw(6)<<a[i];
    cout<<endl;
}
double mult(double b[],int n)
{
    double p=1.0;
    for(int i=0;i<n;i++) p*=b[i];
    return p;
}
```

```

int main()
{
double a[7],b[5],c[9],d[9];
input(a,7);input(b,5);input(c,9);input(d,9);
output(a,7);output(b,5);output(c,9);output(d,9);
double p;
p=(mult(a,7)+log(mult(b,5)))/(sin(mult(c,9))*sqrt(mult(d,9)));
cout<<"p="<<p<<endl;
}

```

```

C:\WINDOWS\system32\cmd.exe
Output mas>
 1 2 3 4 5 6 7
Output mas>
 1 2 3 4 5
Output mas>
 1 2 3 4 5 6 7 8 9
Output mas>
 1 2 3 4 5 6 7 8 9
p=10.5602
Для продолжения нажмите любую клавишу . . .

```

Для целочисленных матриц A и B размерностью 10\*10 сформировать одномерные массивы P и Q, состоящие из числа элементов каждой строки матрицы, кратных 5, и найти наибольшие из них.

```
#include <iostream>
#include <iomanip>
using namespace std;
typedef int matr[10][10];
typedef int mas[10];
void input(matr a, int n, int m)
{
    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)    a[i][j]=rand()/100-100;
}
void output ( matr a, int n, int m)
{
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<m;j++) cout<<setw(6)<<a[i][j];
        cout<<endl;
    }
}
```

```
void form(matr a,mas b,int n, int m)
{
    for(int i=0;i<n;i++)
    {
        b[i]=0;
        for(int j=0;j<m;j++)
            if(a[i][j] % 5==0)b[i]++;
    }
}
int max(mas b,int n)
{
    int maxx=b[0];
    for(int i=0;i<n;i++)
        if(b[i]>maxx)maxx=b[i];
    return maxx;
}
void output_mas(mas a, int m)
{
    for(int i=0;i<m;i++) cout<<setw(6)<<a[i];
    cout<<endl;
}
```

```
int main()
```

```
{
```

```
    matr a,b;
```

```
    mas p,q;
```

```
    input(a,10,10);
```

```
    cout<<"Output matr A"<<endl;
```

```
    output(a,10,10);
```

```
    input(b,10,10);
```

```
    cout<<"Output matr B"<<endl;
```

```
    output(b,10,10);
```

```
    form(a,p,10,10);
```

```
    form(b,q,10,10);
```

```
    cout<<"massiv P"<<endl;
```

```
    output_mas(p,10);
```

```
    cout<<"massiv Q"<<endl;
```

```
    output_mas(q,10);
```

```
    cout<<"maxim P="<<max(p,10)<<"   maxim Q="<<max(q,10)<<endl;
```

```
}
```

```
C:\WINDOWS\system32\cmd.exe
Output matr A>
-100  84  -37  165  91  57  14  193  169  144
-43  181  132  68  -1  -96  -71  19  -52  -46
223  46  -61  -99  -98  23  74  87  97  98
-46  117  47  15  -82  99  156  162  70  -2
187  138  213  203  76  -54  51  -23  182  -32
155  176  226  227  100  28  -13  -3  175  -93
23  -70  121  -82  -98  201  -10  -11  92  126
174  138  58  -33  143  53  50  211  143  -65
96  26  140  99  87  18  -51  -27  39  163
69  224  146  13  -45  115  61  -80  129  65
Output matr B>
-52  211  -54  196  127  -1  39  -77  216  123
-50  187  169  90  -38  -42  167  55  -50  65
139  32  -9  86  123  147  136  55  -60  20
173  -89  69  117  39  -66  211  201  80  13
54  22  177  3  49  -11  222  -3  142  85
-36  149  175  30  64  191  -91  225  87  -84
74  -37  176  105  115  -36  175  -60  -64  143
2  208  -7  10  -55  140  173  131  96  144
-18  -53  -100  -81  164  179  -31  -63  81  -96
-63  48  146  124  78  -76  43  -34  128  -5
massiv P
2  0  0  2  0  3  2  2  1  4
massiv Q
0  5  3  1  1  3  4  3  1  1
maxim P=4   maxim Q=5
Для продолжения нажмите любую клавишу . . .
```

Составить программу с использованием подпрограмм с параметрами для:

1) Ввода массива  $A(10,10)$

2) Формирования одномерного массива  $P$ , в котором элемент равен количеству отрицательных элементов каждого столбца матрицы  $A$

3) Поиска минимального элемента в массиве  $P$

Предусмотреть вывод результатов в следующем виде:

$A_{1,1}=\dots$        $A_{1,2}=\dots$        $A_{1,3}=\dots$       .....  $A_{1,10}=\dots$

$A_{2,1}=\dots$        $A_{2,2}=\dots$        $A_{2,3}=\dots$       .....  $A_{2,10}=\dots$

.....  
 $A_{10,1}=\dots$        $A_{10,2}=\dots$        $A_{10,3}=\dots$       .....  $A_{10,10}=\dots$

$P_1=\dots$      $P_2=\dots$     .....  $P_{10}=\dots$

$P_{\min}=\dots$