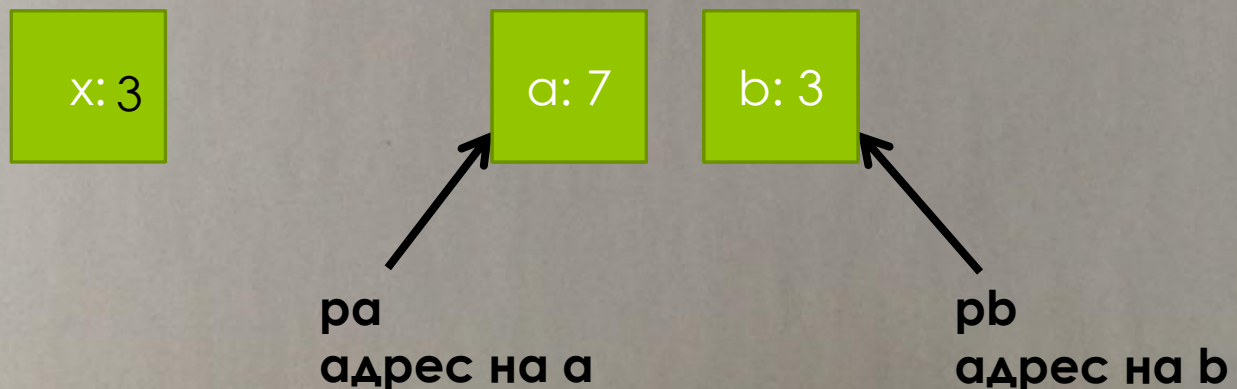




ОСНОВНИ ЗАДАЧИ

гл.ас. д-р. Нора Ангелова

```
#include <iostream.h>
int main()
{ int a = 3, b = 7, x;
  int *pa = &a, *pb = &b;
  cout << *pa << " " << *pb << endl; 3 7
  x = *pa;
  *pa = *pb;
  *pb = x;
  cout << *pa << " " << *pb << endl; 7 3
  return 0;
```



```
#include <iostream.h>
```

```
int main()
```

```
{ int a = 5, b = 4;
```

```
  int *pa = &a, *pb = &b, *p;
```

```
  cout << *pa << " " << *pb << endl;
```

5 4

```
  p = pa;
```

```
  pa = pb;
```

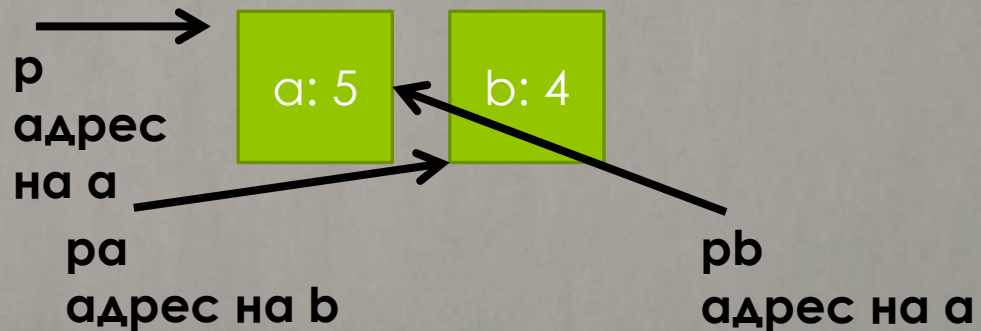
```
  pb = p;
```

4 5

```
  cout << *pa << " " << *pb << endl;
```

```
  return 0;
```

```
}
```



```
#include <iostream.h>
int main()
{ int a, b;
  int* const p = &a;
  *p = 5;
  p = &b; error
  return 0;
}
```

```
#include <iostream.h>
int main()
{ int a = 5, b;
  const int* const p = &a;
  cout << *p << endl;
  *p = 10; error
  p = &b; error
  return 0;
}
```

Низове

(като масиви от символи)

Низ – редица от символи, завършваща с ' $\backslash 0$ ';

Предимство:

- Не е необходимо да се пази дължината на низа

Низове

(като масиви от символи)

Задача.

Да се напише програма, която създава нов обърнат низ.

```
char str[20] = "123454321";  
char reverseStr[20];  
int length = strlen(str);  
  
for(int i=0; i < length; i++) {  
    reverseStr[i] = str[length - i - 1];  
}  
reverseStr[length] = '\0';
```

Низове

(като масиви от символи)

Задача.

Да се напише програма, която проверява дали низ е палиндром. Дължината на низа е ≤ 19 .

```
if (!strcmp(str, reverseStr)) {  
    cout << "Palindrome";  
} else {  
    cout << "Not Palindrome";  
}
```


Низове

(КАТО МАСИВИ ОТ СИМВОЛИ)

Задача. (Вариант 2)

```
char str[20] = "123454321";
int length = strlen(str);

bool isPalindrome = true;
for(int i=0; i<length/2; i++) {
    if (str[i] != str[length-i-1]) {
        isPalindrome = false;
        break;
    }
}

if (isPalindrome) {
    cout << "Palindrome";
} else {
    cout << "Not Palindrome";
}
```

Низове

(като масиви от символи)

Задача.

Символен низ е съставен единствено от малки латински букви. Да се напише програма, която намира и извежда на екрана броя на срещанията на всяка от буквите на низа.

```
#include <iostream.h>
#include <string.h>
const int MAX_SIZE = 100;
int main()
{ char s[MAX_SIZE];
  unsigned int histogram[26] = {0},
              i = 0;
  cout << "s= ";
  cin >> s;
  for(i = 0; i < strlen(s); i++)
    histogram[s[i]-'a']++;
  for(i = 0; i < 26; i++)
    if(histogram[i] > 0)
      cout << (char)('a'+i) << ": "
           << histogram[i] << endl;
  return 0;
}
```

Кой е по-по-най

- Размяна на стойностите на две целочислени променливи. Не може да се използва трета променлива

$a = a + b;$

$b = a - b;$

$a = a - b;$

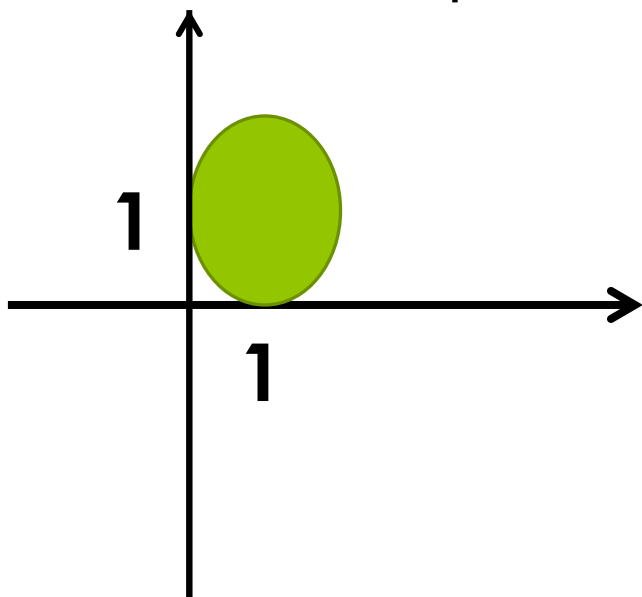
Кой е по-по-най

- Дадено е трицифрено число. Да се изведат цифрите на единиците, десетиците и стотиците. Извеждането да стане на отделни редове в табличен вид.

```
int a ; cin >> a; // а е трицифрено число
short s, d, e;
s = a/100;
d = a/10%10;
e = a%10;
cout << setw(10) << "стотици:" << setw(5) << s << endl;
cout << setw(10) << "десетици:" << setw(5) << d << endl;
cout << setw(10) << "единици:" << setw(5) << e << endl;
```

Кой е по-по-най

- Дали точка принадлежи на областта



$$(x-1)*(x-1)+(y-1)*(y-1) \leq \text{radius}*\text{radius}$$

Кой е по-по-най

- Да се напише програма, която извежда всички трицифрени числа abc , за които е в сила:

$$a + c = b.$$

Пример: 891

```
for(int a=1; a<=9; a++)  
    for(int c = 0; c <= 9-a; c++)  
    {  
        int b = a + c;  
        cout << a << b << c << " ";  
    }
```

Кой е по-по-най

- Дадени са две редици от числа. Да се напише програма, която определя колко пъти първата редица се съдържа във втората.

```
//...
```

```
int br = 0;
if (n <= m) {
    for(int i=0; i<m-n; i++) {
        int j = -1;
        do
            j++;
        while(a[j]==b[i+j] && j<n-1)
            if (a[j] == b[j+i]) br++;
    }
}
```

```
cout << "Първата редица се съдържа във втората " << br
<< "пъти" << endl;
```




```
cout << “Край”;
```