

# Низове

гл.ас. д-р. Нора Ангелова

# Низове

(като масиви от символи)

Задача.

Да се напише програмен фрагмент, който 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 програмен фрагмент, който проверява дали низ е палиндром. Дължината на низа е  $\leq 19$ .

```
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)

```
char str1[10] = "123456789";  
char str2[4] = "abc";  
strcpy(str1, str2);  
cout << str1;
```

Изход:

abc

# Низове

(като масиви от символи)

```
char str1[10] = "0123456789";  
char str2[4] = "abc";  
strcpy(str1, str2);  
cout << str1;
```

Изход:

ERROR – const char[11]

*\*Не работи в повечето версии на езика*

# 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.

\* Не използвайте извикване на функция (strlen и други) във for цикъл.

```
#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;
}
```



# Низове

(като масиви от символи)

- Едномерен масив от символи.
- Името на променлива от тип низ е **константен** указател.

```
char str[7] = "123456";
```

```
while(*str) {  
    cout << *str << " ";  
    str++;  
} // ERROR
```

# Низове

(като масиви от символи)

- Едномерен масив от символи.
- Името на променлива от тип низ е **константен** указател.

```
char str[7] = "abcdef";
```

```
char *p = str;
```

```
while(*p) {  
    cout << *p << " ";  
    p++;  
}
```

```
// указателят str е константен и не може да  
// се променя стойността му
```

```
// ВАЖИ И ЗА ВСЕКИ ДРУГ МАСИВ
```

```
// a b c d e f
```

# Низове

(като масиви от символи)

- Едномерен масив от символи.
- Указател към символ от низа.

```
char str[7] = "abcdef";  
char *p = str;
```

```
cout << p+3; // def
```

Низове  
(като указатели от тип char)

# Низове

(като указател от тип char)

```
char *str = "123456";
```

```
while(*str) {  
    cout << *str << " ";  
    str++;  
}
```

// 1 2 3 4 5 6

# Низове

(като указател от тип char)

\* Не е възможно въвеждане на стойност от тип указател.

```
char *str = "123456";
```

не може да бъде заменено от

```
char *str;  
cin >> str; // ERROR  
(getline, ...)
```

# Низове

(като указател от тип char)

```
char str[7] = "123456";  
*(str+1) = '9';  
str[1] = '9';  
cout << str;
```

Изход:

193456

```
char *str2 = "123456";  
*(str2+1) = '9';  
str2[1] = '9';  
cout << str2;  
// ERROR
```

Изход:

ERROR

ERROR

# Задача

Дадена е квадратна матрица от низове. Да се напише програмен фрагмент, който намира броя на палиндромите под главния диагонал (заедно с него).

```
//...
int br = 0;
char a[MAX_SIZE][MAX_SIZE][MAX_SIZE2];
// Въвеждане на елементите на матрицата
for(int rowIndex = 0; rowIndex < MAX_SIZE; rowIndex++){
    for(int colIndex = 0; colIndex <= rowIndex; colIndex++) {
        // Дължина на низа
        int len = strlen(a[rowIndex][colIndex]);

        // Създаване на обрнат низ
        char revStr[MAX_SIZE2];
        for(int strIndex = len-1; strIndex >= 0; strIndex--) {
            revStr[len - strIndex - 1] = a[rowIndex][colIndex][strIndex];
        }

        revStr[len] = '\0';

        // Сравнение с обрнатия низ
        if (!strcmp(a[rowIndex][colIndex], revStr)) {
            br++;
        }
    }
}
```



## Задача

- Да се напише програмен фрагмент, който намира стойността на правилно записан числов израз без скоби и без приоритет на операторите  $+$ ,  $-$ ,  $*$  и  $/$ .  
Пресмятането да завършва след въвеждане на знака  $=$ .

Например  $3+4*5$

# Задача

```
char op = '+';
double result = 0.0;

do {
    double arg;
    cin >> arg;

    switch (op) {
        case '+': result = result + arg; break;
        case '-': result = result - arg; break;
        case '*': result = result * arg; break;
        case '/': if (arg != 0) {
                    result = result / arg;
                } else {
                    cout << "Error!\n ";
                    return 1;
                }
    }

    cin >> op;
} while (op != '=');

cout << result << endl;
```

Край