

6.C++ Строки.

Специальный класс string

Для его работы необходимо в начале программы подключить заголовочный файл string:

```
#include <string>
```

В отличие от типа char, string является классом. Более подробно об классах я расскажу позднее, сейчас вам достаточно знать, что классы содержат в себе сразу несколько вещей: переменные, константы и функции для работы с переменными.

Теперь чтоб создать строку достаточно написать:

string s;

Для записи в строку можно использовать оператор =

s="Hello";

ПРИМЕР 1. Дана строка s. Вывести длину слова и пятый символ.

```
#include <iostream>
using namespace std;
int main()
{
    string s;
    cout<< "введите строку: ";
    cin>>s;
    int dlina=s.length();
    char k=s[4];           // строка начинается с нулевого символа
    cout<<dlina <<" " <<k;

    system("pause");
    return 0;
}
```

Но пока вы воспользовались только одной прелестью строк: отсутствием необходимости задавать ее размер. Но кроме этого существует множество функций для работы со строками.

- *s.append(str)* - добавляет в конец строки строку str. Можно писать как *s.append(переменная)*, так и *s.append("строка")*;
- *s.assign(str)* - присваивает строке s значение строки str. Аналогично записи *s=str*;
- *int i=s.begin()* - записывает в i индекс первого элемента строки
- *int i=s.end()* - аналогично, но последнего
- *s.clear()* - как следует из названия, очищает строку. Т.е. удаляет все элементы в ней
- *s.compare(str)* -сравнивает строку s со строкой str и возвращает 0 в случае совпадения (на самом деле сравнивает коды символов и возвращает их разность)
- *s.copу(куда, сколько, начиная с какого)* - копирует из строки s в куда (там может быть как строка типа string, так и строка типа

char). Последние 2 параметра не обязательные (можно использовать функцию с 1,2 или 3 параметрами)

- *bool b=s.empty()* - если строка пуста, возвращает true, иначе false
- *s.erase(откуда, сколько)* удаляет n элементов с заданной позиции
- *s.find(str,позиция)* - ищет строку str начиная с заданной позиции
- *s.insert(позиция,str, начиная, beg, count)* - вставляет в строку s начиная с заданной позиции часть строки str начиная с позиции beg и вставляя count символов
- *int len=s.length()* - записывает в len длину строки
- *s.push_back(symbol)* - добавляет в конец строки символ
- *s.replace(index, n,str)* - берет n первых символов из str и заменяет символы строки s на них, начиная с позиции index
- *str=s.substr(n,m)* - возвращает m символов начиная с позиции n
- *s.swap(str)* меняет содержимое s и str местами.
- *s.size()* - возвращает число элементов в строке.

Вот собственно большинство необходимых функций для работы со строками в с++.

[<http://adorning.ru/2010/01/10/c-urok-6-stroki/>]

[http://dovital.narod.ru/Lr_SiAOD7.pdf]

Задание 6.1. Дана строка (<100). Найти количество слов в строке. (разделитель – пробел, конец строки-точка)

Задание 6.2. Дана строка (<100). Сколько в строке букв «М»? Сколько слов начинаются на букву «М»?

Задание 6.3. Дана строка (<100). Отсортировать все символы в алфавитном порядке.

Задание 6.4. Дана строка (<100). Выписать все слова в алфавитном порядке.

Задание 6.5. Дана строка (<100). Найти наибольшее число в строке. Например,

Вход	Выход
Ек67па55па099лоо	99

Задание 6.6. Дана строка (<100) из кириллических (или латинских) символов. Заменить каждую букву через три по алфавиту. Например,

Вход	Выход
апроя	гтусв

Задание 6.7. Дана строка (< 100) из кириллических (или латинских) символов. Найти наиболее часто встречающийся символ.

Задание 6.8. Дана строка (< 100) из кириллических (или латинских) символов. Удалить все повторные вхождения букв. Оставив только первое вхождение.

Задание 6.9. Дано натуральное число (< 100000000000000). Является ли оно палиндромом? (Y,N)

Задание 6.10. Дано натуральное число в десятичной системе (< 1000000). Является ли оно палиндромом в шестнадцатеричной?