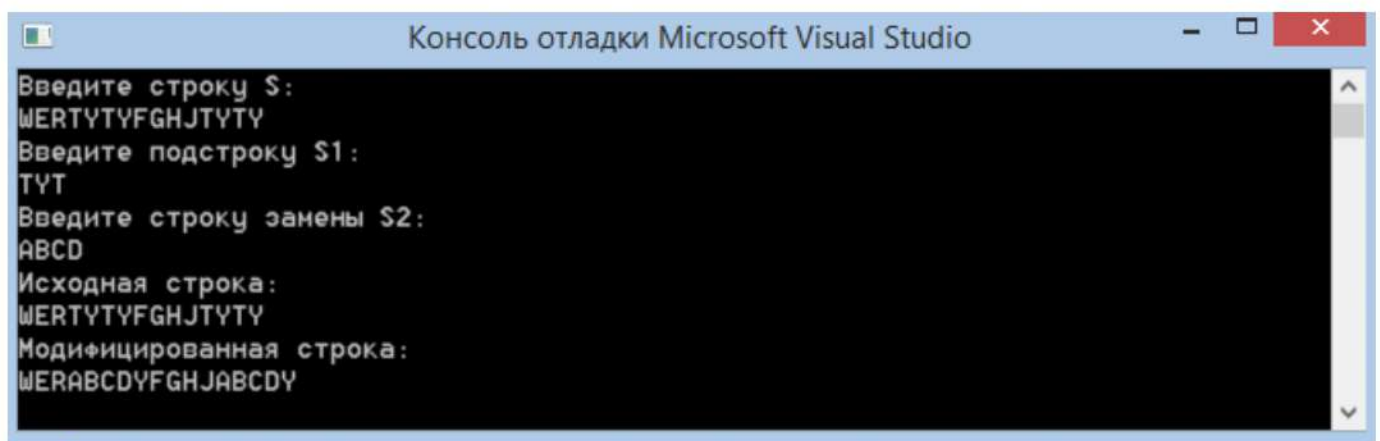


Строки.

Пример 1. Поиск и замена символов. Даны строки  $S$ ,  $S1$  и  $S2$ . Заменить в строке  $S$  все вхождения строки  $S1$  на строку  $S2$ .

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    setlocale(LC_ALL, "rus");
    string S, S1, S2, Scopy;
    cout << "Введите строку S:" << endl;
    getline(cin, S);
    Scopy = S; // S изменится, поэтому сохраняем её в Scopy
    cout << "Введите подстроку S1:" << endl;
    getline(cin, S1);
    cout << "Введите строку замены S2:" << endl;
    getline(cin, S2);
    size_t k = S1.size(); // k - число символов подстроки S1
    while (S.find(S1) != string::npos)
    {
        size_t i = S.find(S1); // i - начальная позиция подстроки S1 в строке S
        S.replace(i, k, S2); // замена k символов подстроки S1, начиная с
                             // позиции i, на символы строки замены S2
    }
    cout << "Исходная строка:\n"
         << Scopy << endl
         << "Модифицированная строка:\n"
         << S << endl;
    return 0;
}
```



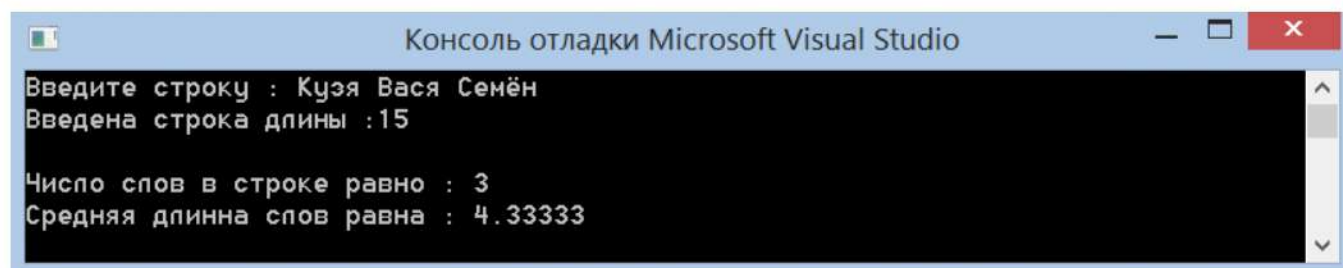
Консоль отладки Microsoft Visual Studio

```
Введите строку S:
WERTYTYFGHJTYTY
Введите подстроку S1:
TYT
Введите строку замены S2:
ABCD
Исходная строка:
WERTYTYFGHJTYTY
Модифицированная строка:
WERABCDYFGHJABCDY
```

**Пример 2.** Найти среднюю длину слов в введенной строке. Слова в строке отделяются одним пробелом.

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

int main()
{
    setlocale(LC_ALL, "Rus");
    char str[256]; // объявляем строку
    cout << "Введите строку : "; cin.getline(str, 256);
    int lstr = strlen(str);
    cout << "Введена строка длины :" << lstr << endl;
    double average, summa, count; // объявляем переменные для
    // средней длины, суммы длин слов и кол-ва слов
    summa = 0;
    count = 0;
    for (int i = 0; i < lstr; i++)
    {
        int j = i;
        while ( j < lstr && str[j] != ' ' )
        {
            summa += 1; // считаем сумму длин слов
            j++;
        }
        count += 1; // увеличиваем кол-во слов в строке
        i = j;
    }
    average = summa / count; // считаем среднюю длину
    cout << endl;
    cout << "Число слов в строке равно : " << count << endl;
    cout << "Средняя длинна слов равна : " << average << endl;
    return 0;
}
```



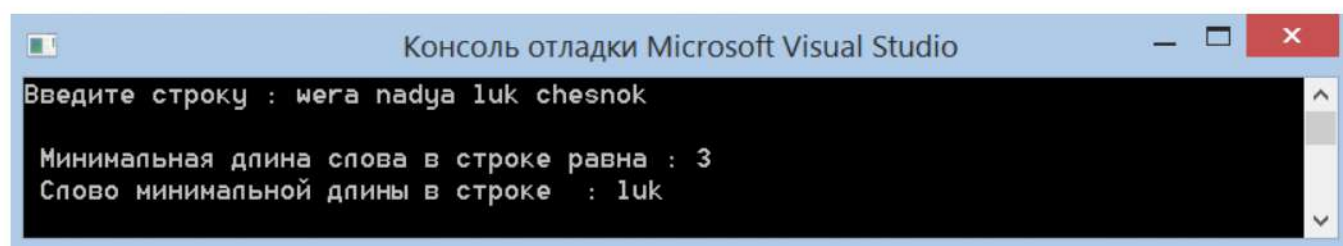
Консоль отладки Microsoft Visual Studio

```
Введите строку : Кузя Вася Семён
Введена строка длины :15

Число слов в строке равно : 3
Средняя длинна слов равна : 4.33333
```

**Пример 3.** Дана строка слов, разделенных пробелами. Вывести на экран самое короткое слово строки. Слова в строке отделяются одним пробелом.

```
#include <iostream>
#include <string.h>
using namespace std;
int main()
{
    setlocale(LC_ALL, "Rus");
    char str[256]; // объявляем строку
    char word[20], wordm[20] ; // объявляем слово
    cout << "Введите строку : "; cin.getline(str, 256);
    int lstr = strlen(str); // длина введенной строки
    double lword, lmin; // длина слова и минимальная длина слова
    lmin = 256;
    for (int i = 0; i < lstr; i++)
    {
        lword=0;
        int j = i;
        while ( j < lstr  &&  str[j] != ' ')
        {
            lword += 1; // считаем длину слова
            word[j-i] = str[j]; //формируем слово
            j++;
        }
        word[j-i] = '\0';
        i = j;
        if (lword < lmin)
        {
            lmin = lword;
            for (int k = 0; k<=strlen(word);k++)
                wordm[k]= word[k]; // запоминаем текущее минимальное слово
        }
    }
    cout << endl;
    cout << "Минимальная длина слова в строке равна:"<< lmin<< endl;
    cout << "Слово минимальной длины в строке: " << wordm << endl;
    return 0;
}
```



```
Консоль отладки Microsoft Visual Studio
Введите строку : wega nadya luk chesnok

Минимальная длина слова в строке равна : 3
Слово минимальной длины в строке : luk
```



**Пример 4.** Дана строка символов `s1`, состоящая максимум из 80 символов, представляющая собой код, составленный по следующему правилу: каждая буква заменяется на её порядковый номер (двухзначное число) в алфавите, т.е. '01' – 'А', '02' – 'В' и т.д. . Напишите программу, которая декодирует строку. Например, для кода 01020103 получим слово 'АВАС'

**Пояснения.** При поэлементном заполнении символьного массива `str` (алфавит А ... Z) и `s2` (декодированная строка) необходимо помнить, что последним символом после ввода символов строки должен быть нуль-символ `'\0'`. Следует обратить внимание на способ изменения символьной переменной `s` (буква алфавита): несмотря на то, что переменная `s` имеет тип `char`, для ее изменения используется оператор инкремента `++`, что позволяет перебирать буквы алфавита:

```
char s = 'A'; str[0] = '_';
for (int i = 1; i < n; i++, s++) str[i] = s;

str[n] = '\0';
```

Кроме того, функция `atoi(ss)` преобразует строку `ss` в число целого типа (`int`), что позволяет код буквы заменить буквой алфавита.

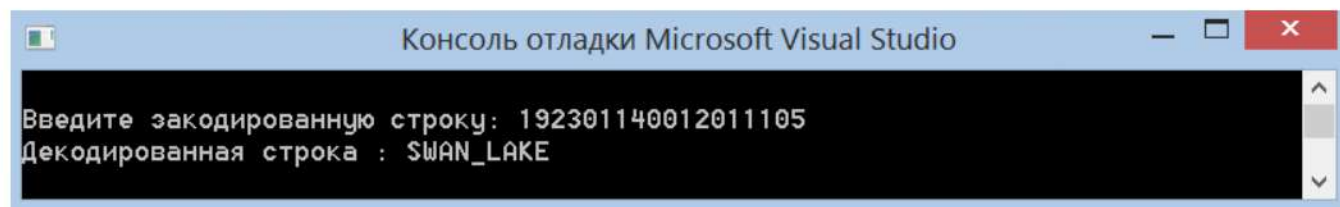
```
#include <iostream>
#include <cstring>
#include <cstdlib>
using namespace std;
int main() {
    setlocale(LC_ALL, "rus");
    char str[30], s1[80], s2[40] = "_";
    int n = 27;
    int j = 0;
    cout << endl;
    cout << "Введите закодированную строку: ";
                                     //задаём закодированную строку из цифр
    cin.getline(s1, 79);
    str[0] = '_'; char s = 'A';
    for (int i = 1; i < n; i++, s++) //в строку str помещаем буквы
                                     //А...Z
        str[i] = s;
    str[n] = '\0'; // последний символ строки str - нуль-символ '\0'
    int n1 = strlen(s1), i1;
    for (int i = 0; i < n1; i += 2) //производим декодирование строки s1
                                     //в строку s2
    {
        char ss[3] = { '0', '0' }; ss[0] = s1[i]; // ;
        ss[1] = s1[i + 1]; ss[2] = '\0';
        i1 = atoi(ss);
        if (i1 < 27) s2[j] = str[i1]; else s2[j] = '*';
        j++;
    }
}
```

```

s2[j] = '\\0'; // последний символ строки s2 - нуль-символ '\\0'
cout << "Декодированная строка : " << s2 << endl;
return 0;
}

```

Результат



**Пример 5.** Дана строка символов S с одним пробелом между словами. Напишите программу, которая утроит каждый пробел между словами.

```

#include <iostream>
#include <cstring>
#include <stdio.h>
using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    char s[1000];
    cout << "Введите строку: "; //вводим строку
    cin.getline(s, 100);
    int n = strlen(s);
    for (int j = 0; j < n; j++)
    {
        if (s[j] == ' ') //ищем пробелы
        {
            n += 2;
            for (int i = n; i > j; i--) // сдвиг части строки
                //вправо на 2 позиции
                s[i] = s[i - 2];
            s[j + 1] = ' ';
            s[j + 2] = ' ';
            j += 2;
        }
    }
    cout << "Преобразованная строка : " << s << endl;
    return 0;
}

```

```
Консоль отладки Microsoft Visual Studio
Введите строку: Balet "Swan lake"!
Преобразованная строка : Balet  "Swan  lake"!
```

**Пример 6.** Введена строка символов. Удалите из нее все кратные рядом стоящие одинаковые символы, оставив по одному. Например: BBAAAALEET=>BALET.

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

using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    char s[100];
    cout << "Введите строку: "; //вводим строку
    cin.getline(s, 100);
    int n = strlen(s), i = 0, j = 0; //считаем кол-во символов
    while (i < n)
    {
        if (s[i] == s[i + 1]) //проверяем условие
        {
            for (j = i; j < n; j++) //удаляем символы
                s[j] = s[j + 1];
            n--;
        }
        else i++;
    }
    cout << "Строка после удаления повторяющихся подряд символов: ";
    puts(s); //выводим строку
}
```

```
Консоль отладки Microsoft Visual Studio
Введите строку: BBAAALLLEET
Строка после удаления повторяющихся подряд символов: BALET
```