

**C++ DASTURLASH TILIDA SATRLI IFODALAR BILAN  
ISHLASH RAБOTA CO CTPOKAMI B C++  
WORKING WITH STRINGS IN C++**

**Shermatova Xilola Mirzayevna**

*Farg‘ona davlat universiteti Axborot texnologiyalari kafedrası dotsenti*

[shermatovahilola1978@gmail.com](mailto:shermatovahilola1978@gmail.com)

**Qurbonaliyeva Feruzaxon Abduvohid qizi**

*Farg‘ona davlat universiteti Axborot tizimlari va  
texnologiyalari yo‘nalishi 1-kurs talabasi*

**Annotatsiya.** Ushbu maqolada C++ dasturlash tilida satrli ifodalar bilan ishlashning asosiy tamoyillari batafsil ko‘rib chiqiladi. Satrlarni saqlash va qayta ishlash usullari, jumladan C-style strings va zamonaviy `std::string` sinfining imkoniyatlari, ularning afzallik va kamchiliklari haqida tushuncha beriladi. Maqolada C++ dasturlash tilida satrlar ustida bajariladigan asosiy operatsiyalar, jumladan nusxalash, qo‘shish, uzunligini aniqlash, taqqoslash va qidirish kabi amallar misollar orqali tushuntiriladi. Har bir usulning afzallik va kamchiliklari muhokama qilinib, amaliy dasturlashda eng samarali yondashuvlar tavsiya etiladi. Shuningdek, maqolada satrlar bilan ishlash jarayonida yuzaga kelishi mumkin bo‘lgan xatoliklar va ularni oldini olish bo‘yicha maslahatlar ham beriladi. Ushbu maqola C++ dasturlash tilini o‘rganayotgan yoki amaliy loyihalarda satrli ma‘lumotlar bilan ishlayotgan dasturchilar uchun foydali bo‘ladi.

**Аннотация.** В данной статье подробно рассматриваются основные принципы работы со строковыми выражениями в языке программирования C++. Описаны методы хранения и обработки строк, включая C-style strings и современные возможности класса `std::string`, а также их преимущества и недостатки. В статье рассматриваются основные операции над строками в C++, такие как копирование, конкатенация, определение длины, сравнение и поиск. Все методы объясняются на примерах, обсуждаются их плюсы и минусы, а также предлагаются наиболее эффективные подходы в практическом программировании. Кроме того, в статье даются советы по предотвращению возможных ошибок при работе со строками. Данный материал будет полезен как изучающим язык C++, так и разработчикам, работающим с текстовыми данными в своих проектах.

**Annotation.** This article provides a detailed overview of the fundamental principles of working with string expressions in the C++ programming language. It covers methods for storing and processing strings, including C-style strings and the modern `std::string` class, highlighting their advantages and disadvantages. The article explores essential string operations in C++, such as copying, concatenation, length determination,

comparison, and searching. Each method is explained with examples, discussing its pros and cons, and recommending the most efficient approaches for practical programming. Additionally, the article offers advice on preventing potential errors when working with strings. This material will be useful for both those learning C++ and developers working with textual data in their projects.

**Kalit soʻzlar:** C++ dasturlash, satrli ifodalar, C-style strings, std::string, matnli maʼlumotlar, satr operatsiyalari, nusxalash, qoʻshish, taqqoslash, qidirish, dasturlash amaliyoti.

**Ключевые слова:** Программирование на C++, строковые выражения, C-style strings, std::string, текстовые данные, операции со строками, копирование, конкатенация, сравнение, поиск, практическое программирование.

**Keywords:** C++ programming, string expressions, C-style strings, std::string, textual data, string operations, copying, concatenation, comparison, searching, practical programming.

## Kirish

Satrli ifodalar dasturlash tillarida matnli maʼlumotlar bilan ishlashning muhim qismi hisoblanadi. Matnlarni saqlash, tahlil qilish va manipulyatsiya qilish dasturlashning ajralmas qismi boʻlib, turli dasturlash tillarida bu jarayonlar har xil amalga oshiriladi. C++ dasturlash tilida satrlar bilan ishlash uchun ikkita asosiy yondashuv mavjud: anʼanaviy C-style strings va zamonaviy std::string sinfi.

Anʼanaviy C-style satrlari oddiy char massivlari koʻrinishida boʻlib, oxirida \0 belgisiga ega. Ular dasturlashning ilk bosqichlarida ishlatilgan boʻlsa-da, xavfsizlik va foydalanish qulayligi nuqtai nazaridan kamchiliklarga ega. std::string sinfi esa zamonaviy dasturlashda keng qoʻllanib, satrlar ustida qulay va xavfsiz operatsiyalar bajarishga imkon beradi. Ushbu sinf xotira boshqaruvini avtomatlashtirib, dasturchilarning ishini ancha osonlashtiradi.

Satrlar bilan ishlash dasturlashning koʻplab sohalarida, jumladan veb dasturlash, maʼlumotlarni tahlil qilish, foydalanuvchi interfeyslari bilan ishlash kabi jarayonlarda keng qoʻllaniladi. Ushbu maqolada C++ tilida satrlar bilan ishlash usullari, satrlar ustida bajariladigan operatsiyalar va ularning dasturda qoʻllanilishiga oid misollar batafsil bayon etiladi. Shu bilan birga, satrlar bilan ishlashda samaradorlik va xavfsizlikni taʼminlashga oid tavsiyalar ham beriladi.

## C-style Strings

C-style satrlar aslida char massivlari boʻlib, oxirida \0 belgisi bilan tugaydi. Quyida C-style satr bilan ishlashga misol keltirilgan:

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
```

```

char str1[] = "Salom";
char str2[10];
strcpy(str2, str1); // Satrni nusxalash
strcat(str2, "! Dunyoga"); // Satrlarni qo'shish
cout << "Natija: " << str2 << endl;
cout << "Satr uzunligi: " << strlen(str2) << endl;
return 0; }

```

Bu usuldan foydalanish xavfsizlik jihatidan ba'zi kamchiliklarga ega bo'lishi mumkin, shuning uchun zamonaviy **std::string** dan foydalanish tavsiya etiladi.

### **std::string** sinfi

C++ STL (Standard Template Library) tarkibidagi **std::string** sinfi satrlar bilan ishlashni ancha qulaylashtiradi.

#### **Satr yaratish va chop etish**

```

#include <iostream>
#include <string>
using namespace std;
int main() {
    string str = "Salom, C++!";
    cout << str << endl;
    return 0; }

```

#### **Satr uzunligini aniqlash**

```
std::cout << "Satr uzunligi: " << str.length() << std::endl;
```

#### **Satrlarni qo'shish**

```

std::string str1 = "Salom";
std::string str2 = "Dunyo";
std::string natija = str1 + str2;
std::cout << natija << std::endl;

```

#### **Satrni taqqoslash**

```

if (str1 == str2) {
    std::cout << "Satrlar teng!" << std::endl;
} else {
    std::cout << "Satrlar har xil!" << std::endl;
}

```

#### **Satr chida qidirish**

```

std::size_t pos = str.find("C++");
if (pos != std::string::npos) {
    std::cout << "Topildi! Pozitsiya: " << pos << std::endl;
} else {
    std::cout << "Topilmadi!" << std::endl;
}

```

}

### **Xulosa**

C++ dasturlash tilida satrlar bilan ishlashning turli usullari mavjud. C-style strings oddiy matnlarni saqlash va manipulyatsiya qilish uchun ishlatilgan bo'lsa-da, ular xavfsizlik muammolariga duch keladi. Shu sababli zamonaviy dasturlashda `std::string` sinfidan foydalanish tavsiya etiladi. `std::string` sinfi yordamida satrlarni qulay tarzda yaratish, birlashtirish, taqqoslash va qidirish mumkin. U operatsion tizim va kompilyator bilan mos ravishda ishlaydi hamda qo'shimcha xotira boshqaruvini talab qilmaydi. Satrlar bilan ishlash C++ dasturlarining ajralmas qismi bo'lib, har qanday dasturchi ular bilan samarali ishlashni o'rganishi lozim. Ushbu maqolada keltirilgan misollar va tushunchalar orqali C++ da satrli ifodalar bilan ishlash bo'yicha asosiy bilimlarga ega bo'lish mumkin. Keyingi bosqichda `std::wstring`, `std::stringstream` va boshqa rivojlangan satr ma'lumot tuzilmalari bilan ishlashni ham o'rganish tavsiya etiladi. Shuningdek, dasturlash jarayonida kod samaradorligini oshirish uchun satrlar bilan ishlashning eng yaxshi amaliyotlarini o'zlashtirish muhim ahamiyat kasb etadi.

### **FOYDALANILGAN ADABIYOTLAR:**

1. Bjarne Stroustrup - The C++ Programming Language (4th Edition)
2. Scott Meyers - Effective Modern C++
3. Nicolai M. Josuttis - The C++ Standard Library: A Tutorial and Reference
4. Bruce Eckel - Thinking in C++
5. C++ Reference: <https://cplusplus.com/reference/string/>
6. C++ Standard Library Documentation: <https://en.cppreference.com>