



Клас String

Приклад ANSI-рядку та string

Робота з ANSI-рядком

```
#include <iostream>
#include <cstring> // Використовуємо strcpy()
#include <cstdlib> // Використовуємо malloc()

int main(int argc, char **argv){
    char CC[17]; // C -string (16 символів + NULL
termination)
    char *CC2; // C-string. Память не виділена

    strcpy(CC, "This is a string"); // ініціалізуємо CC
    CC2 = (char *) malloc(17); // Виділяємо пам'ять під
рядок.

    strcpy(CC2, CC); //strcpy(CC2, "This is a string");

    cout << CC << endl;
    cout << CC2 << endl;
    // де тут помилка?
}
```

Робота з класом string

```
#include <iostream>
#include <string>

int main(int argc, char **argv){

    std::string SS; // C++ STL рядок
    std::string SS2; // C++ STL рядок

    SS = "This is a string"; //
ініціалізуємо
    SS2 = SS; // копіюємо

    std::cout << SS << endl;
    std::cout << SS2 << endl;
}
```

Клас string

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

Конструктори:

1. `string` x("Literal");
 `string` sVar1("Програмування на C++");
2. `char` cVar[10]= "Hello!";
 `string` sVar2(cVar);
3. `string` sVar3(5, 'a'); // 5 символів 'a': "aaaaa"
4. `string` Var1 ("String Constructor");
 `string` sVar4(Var1, 8); // Ініціалізує рядок з 8-го символу до кінця рядка //Var1:
 "Constructor"
5. `string` sVar5(Var1.begin(), Var1.end());

Ініціалізація рядку

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

```
string Var, Var1; // дві змінні типу рядок
Var = "Some string variable"; // одну ініціалізували літералом
Var1.assign(" Some string-to-assign"); // а іншу ось так
```

```
string Var2("vasya"); // а тут конструктором копіювання
```

```
if(Var2) Var1.assign(Var2); // перевірили перед присвоюванням
```

```
Var2 += Var1; // конкатинація через унарний оператор
Var2.append("!!!"); // через append
```

```
Var2.push_back('?'); // додали символ до кінця
```

Вставка та заміна в рядку

```
string x{"Programming on C++"}; // ініціалізація
string y(" C and "), z; // декларація
cout<<"My favourite laguage is "<<y[1]<<"\n";
```

```
x.insert(12,y); // вставка
char p = x[0]; // доступ
x.replace(0,1,"we study here ", 0, 15); // заміна
cout<< x<<endl;
x[14] = tolower(p); // заміна доступом
cout<< x<<endl;
```

```
z = " Python, Java, ";
size_t m = x.size(); // розмір рядку
```

```
x.insert(m, z, 0, 8); // вставка
cout<< x<<endl;
cout<<"but after University...";
x.erase(); // видалення
cout<<"we really know "<<x;
```

My favourite laguage is C
we study here rogramming C and on C++
we study here programming C and on C++
we study here programming C and on C++
Python,
but after University...we really know

Пошук та заміна в рядку

```
string str1 = " find the hypothenous x=3 x=4 x=5 "; // конструктор копіювання
```

```
string expr = "x="; // і тут
```

```
string::size_type m = str1.find(expr); // знайшли потрібний підрядок
```

```
if (m == string::npos ) { // m!=-1 // перевірка того, що знайшли  
    cout<<"no entries";  
}
```

```
string::size_type m2 = str1.rfind(expr); // знайшли справа
```

```
string::size_type m3 = str1.find(expr, m1+1); // пошук з даного місця
```

```
string tmp = str1.substr(m3,m3+3); // віділили підрядок
```

```
str1.replace(m3,2,"y="); // замінили його
```

Перевод в/з рядку з/в числового типу

```
string sVar("GGGGG");  
char* x = svar.c_str();
```

```
string sVar1("LL1234");  
1) int num = stol ( sVar1.c_str(), nullptr);  
2) int num = atoi( sVar1.c_str() );  
3) if(sscanf(sVar1.c_str(), "%d", &i) != 4)  
{  
    int num = std::stoi( sVar1 );  
}  
5) #include <sstream>  
    stringstream tmp(s);  
    int num = 0;  
    tmp >> num;
```

```
1) res = to_string(number); //c++11
```

Або

2)

```
#include <sstream> // ostringstream  
string int2string1(const int& number){  
    ostringstream oss;  
    oss << number;  
    return oss.str();  
}
```

3)

```
string res = int2string1(number);  
//або функцією sprintf :  
const size_t size=255;  
char text_num[size];  
sprintf(text_num, "%d", number);  
string res1(text_num);
```

Обробка файлу з роздільниками

```
string::size_type posBeginIdx, posEndIdx;
string::size_type ipos=0;
string sLine, sValue;
string sKeyWord;
const string sDelim(", ");
string sError;
ifstream myInputFile("DataFile.txt", ios::in);
while(getline(myInputFile, sLine)){
    if(myInputFile.bad()){
        cerr<<"Error with file"; break;
    }
    if( !sLine.empty()){
        posEndIdx = sLine.find_first_of(sDelim);
        if(posEndIdx==string::npos) break;
        sKeyWord = sLine.substr( ipos, posEndIdx ); // Extract word
        posBeginIdx = posEndIdx + 1; // Beginning of next word (after ':')
        cout<<posBeginIdx<<sKeyWord<<posEndIdx;
        ipos++;
    }
}
```