



Laurea triennale in Informatica

modulo (CFU 6) di

Programmazione II e Lab.

prof. Mariarosaria Rizzardi

Centro Direzionale di Napoli – Isola C4

stanza: n. 423 – IV piano Lato Nord

tel.: 081 547 6545

email: mariarosaria.rizzardi@uniparthenope.it

➤ Stringhe in C++

Esempio 0: stringhe mediante array di char, come in C

```
#include <iostream>
using namespace std;
int main()
{   char a='a';   char *A="a";
    return 0;
}
```

stringa
↓
↑

NO!

warning: ISO C++ forbids converting a string constant to 'char*'

```
#include <iostream>
using namespace std;
int main()
{   char a='a';   const char *A="a";
    cout << "a = " << a << "; sizeof(a) = " << sizeof(a) << endl;
    cout << "A = " << A << "; sizeof(A) = " << sizeof(A) << endl;
    return 0;
}
```

OK!

a = a; sizeof(a) = 1
A = a; sizeof(A) = 8

perché?

```
#include <iostream>
using namespace std;
int main()
{   char a='a';   char A[]="a";
    cout << "a = " << a << "; sizeof(a) = " << sizeof(a) << endl;
    cout << "A = " << A << "; sizeof(A) = " << sizeof(A) << endl;
    return 0;
}
```

OK!

a = a; sizeof(a) = 1
A = a; sizeof(A) = 2

perché?

Esempio 1: la classe string in C++

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string mystring;
    mystring = "This is a string";
    cout << mystring << endl;
    ...
    This is a string
}
```

Altre inizializzazioni

string mystring = "This is a string";

string mystring ("This is a string");

string mystring {"This is a string"};

```
...
cout << "'" << mystring[5] << "'" << endl;
cout << "'" << mystring.at(5) << "'" << endl;
...
    'i'
```

1 carattere

```
...
string str;
str.assign(mystring, 5, 4);
cout << "\"" << str << "\"" << endl;
return 0;
    "is a"
```

sottosstringa

This is a string
012345
← 4 →

Esempio 2: operazioni su string

```
#include <iostream>
#include <string>
using namespace std;
int main()
{   string str1, str2;
    str1 = "ABCD";
    str2 = "EFG";
    cout << str1 + ' ' + str2 << endl;
    ...
}
```

concatenazione

ABCD EFG

```
...
if (str1 < str2)
    cout << str1 << endl;
else
    cout << str2 << endl;
return 0;
}
```

confronto

ABCD

Esempio 3: altre operazioni su string

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str1, str2;
    str1 = "ABCD";
    str2 = "EFG";
    str1.insert(2, str2);
    cout << str1 << endl;
    str1.erase(2, 3);
    cout << str1 << endl;
    str1.replace(1, 2, str2);
    cout << str1 << endl;
    int i = str1.find("EF");
    cout << "string::npos = " << (int)string::npos << endl;
    if (i != string::npos)
    {
        str2="EF";
        cout << "\"" << str2 << "\" trovata in str1 a " << i << endl;
        str2.assign(str1, i+str2.length(), str1.length());
        cout << "stringa rimanente: " << str2 << endl;
    }
    return 0;
}
```

insert() → AB**EFG**CD

erase() → ABCD

replace() → AE**FG**D

find() → string::npos = -1
"EF" trovata in str1 a 1
stringa rimanente: GD

posizione dopo la fine della stringa

str1 AEFGD

str2 EF

i 1

str2.length() 2

```
str2.assign(str1, i+str2.length(), str1.length());
```

Copia in **str2** la porzione di **str1** che inizia alla posizione **i+str2.length()=3** e ha lunghezza **str1.length()=5** caratteri (oppure fino alla fine di **str1**, se o **str1** è troppo corta oppure se **str1.length()** è `string::npos`).

str1 AEF**GD**

str2 **GD**

Esempio 4a: main che restituisce il nome del mese (con array statico)

esempio di array frastagliato

```
#include <iostream>
#include <string>
using namespace std;
int main()
{   int n=13;
    string mesi[] = {"Numero di mese non corretto",
                    "Gennaio", "Febbraio", "Marzo", "Aprile",
                    "Maggio", "Giugno", "Luglio", "Agosto",
                    "Settembre", "Ottobre", "Novembre", "Dicembre"};

    int Nmese, m;
    cout << "numero mese = "; cin >> Nmese;
    if (Nmese < 1 || Nmese > 12)
        m=0;
    else
        m=Nmese;
    cout << "mese corrispondente = " << mesi[m] << endl;
    return 0;
}
```

array di stringhe

numero mese = 3
mese corrispondente = Marzo

Esempio 4b: main che restituisce il nome del mese (con array di puntatori)

esempio di array frastagliato

```
#include <iostream>
#include <string>
using namespace std;
int main()
{   int n=13;
    string *mesi = new string[n] {"Numero di mese non corretto",
                                   "Gennaio", "Febbraio", "Marzo", "Aprile",
                                   "Maggio", "Giugno", "Luglio", "Agosto",
                                   "Settembre", "Ottobre", "Novembre", "Dicembre"};

    int Nmese, m;
    cout << "numero mese = "; cin >> Nmese;
    if (Nmese < 1 || Nmese > 12)
        m=0;
    else
        m=Nmese;
    cout << "mese corrispondente = " << mesi[m] << endl;
    return 0;
}
```

allocazione dinamica

array di puntatori a stringhe

numero mese = 3
mese corrispondente = Marzo

Esempio 4c: main che restituisce il nome del mese (con vector di string)

esempio di array frastagliato

```
#include <iostream>
#include <string>
#include <vector>
using namespace std;
int main()
{   int n=13;
    vector <string> mesi = {"Numero di mese non corretto",
                             "Gennaio","Febbraio","Marzo","Aprile",
                             "Maggio","Giugno","Luglio","Agosto",
                             "Settembre","Ottobre","Novembre","Dicembre"};

    int Nmese, m;
    cout << "numero mese = "; cin >> Nmese;
    if (Nmese < 1 || Nmese > 12)
        m=0;
    else
        m=Nmese;
    cout << "mese corrispondente = " << mesi[m] << endl;
    return 0;
}
```

vector di stringhe

numero mese = 3
mese corrispondente = Marzo

Esempio 5: input di stringhe

```
#include <iostream>
#include <string>
#include <limits>
using namespace std;
int main()
```

```
{   string name;
    cout << "Cognome Nome: ";
    cin >> name;
    cout << "Immeso: \"" << name << "\"" << endl << endl;
    cin: si ferma al primo spazio
```

```
// ignora i rimanenti caratteri presenti nello stream di input
cin.ignore(numeric_limits<streamsize>::max(), '\n');
```

```
    cout << "Cognome Nome: ";
    getline(cin, name);
```

```
    cout << "Immeso: \"" << name << "\"" << endl;
    return 0;
```

```
}
```

std::getline (string)

Cognome Nome: Rossi Mario
Immeso: "Rossi"

Cognome Nome: Rossi Mario
Immeso: "Rossi Mario"

getline: si ferma al primo '\n'

<http://www.cplusplus.com/reference/string/string/>

```
string
string::string
string::~string
member functions:
string::append
string::assign
string::at
string::back
string::begin
string::capacity
string::cbegin
string::cend
string::clear
string::compare
string::copy
string::crbegin
string::crend
string::c_str
string::data
string::empty
string::end
string::erase
string::find
string::find_first_not_of
string::find_first_of
string::find_last_not_of
string::find_last_of
string::front
string::get_allocator
string::insert
```

```
string::insert
string::length
string::max_size
string::operator+=
string::operator=
string::operator[]
string::pop_back
string::push_back
string::rbegin
string::rend
string::replace
string::reserve
string::resize
string::rfind
string::shrink_to_fit
string::size
string::substr
string::swap
member constants:
string::npos
non-member overloads:
getline (string)
operator+ (string)
operator<< (string)
operator>> (string)
relational operators (string)
swap (string)
```