



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

The background features a large, faint, circular seal of the University of Naples Federico II. The seal includes the text "1920 - 2020" at the top, "DEGLI STUDI" in the center, "UNIVERSITA' FEDERICO II" around the perimeter, and "100° ANNIVERSARIO" at the bottom. The central part of the seal depicts a shield with a hand holding a torch.

Programmazione in C++:

- **cenni sui namespace**
- **clausola using**

Si è visto che per “implementare” un metodo al di fuori della sua classe bisogna usare:

nome_classe::nome_metodo

Il simbolo **::** è chiamato *scope resolution operator*. Esso informa il compilatore che tale metodo appartiene alla classe **nome_classe**, ovvero che esso è nello *scope* (ambito di visibilità) di **nome_classe**.

namespace **std**

```
#include<iostream>
int main()
{
    std::cout << "sizeof(int) = " << sizeof(int) << std::endl;
    return 0;
}
```

```
#include<iostream>
using namespace std;
int main()
{
    cout << "sizeof(int) = " << sizeof(int) << endl;
    return 0;
}
```

Il *namespace* è una regione dichiarativa il cui obiettivo è “localizzare” i nomi degli identificatori evitando collisioni di nome. Gli elementi dichiarati in un *namespace* sono separati dagli elementi dichiarati in un altro *namespace*.

namespace

La sintassi per creare un *namespace* è:

```
namespace nome {  
    // dichiarazioni  
}
```

Esempio 1a

```
#include<iostream>  
using namespace std;  
  
namespace space1 {  
    void func() { cout << "dentro space1" << endl; }  
}  
  
namespace space2 {  
    void func() { cout << "dentro space2" << endl; }  
}  
  
int main()  
{  
    space1::func();  
    space2::func();  
    return 0;  
}
```

```
dentro space1  
dentro space2
```

namespace

La sintassi per creare un *namespace* è:

```
namespace nome {  
    // dichiarazioni  
}
```

Esempio 1b

```
#include<iostream>  
using namespace std;  
  
namespace space1 {  
    void func() { cout << "dentro space1" << endl; }  
}  
  
namespace space2 {  
    void func() { cout << "dentro space2" << endl; }  
}  
  
using namespace space2;  
  
int main()  
{  
    func();  
    func();  
    return 0;  
}
```

```
dentro space2  
dentro space2
```

using

L'uso di **using** ha due forme generali:

```
using namespace nome;  
using nome::member;
```

La prima si riferisce al namespace che si vuole usare e serve per semplificare il riferimento agli elementi più frequentemente usati del namespace.

La seconda fa riferimento ad un singolo membro del namespace.

Esempio

```
#include<iostream>  
using std::cout;  
int main()  
{  
    cout << "sizeof(int) = " << sizeof(int) << std::endl;  
    return 0;  
}
```

cout è usato senza namespace

endl va usato col namespace