

Argomenti costanti

Pericle Perazzo

`pericle.perazzo@iet.unipi.it`

`http://www.iet.unipi.it/p.perazzo/teaching/`

`http://lettieri.iet.unipi.it/mailman/listinfo/algoritmi_e_basi`

12 aprile 2013



Argomento costante in ingresso

```
int fun1(const string& s) { ... }
```

```
int fun2(const Node* t) { ... }
```

- Aumenta la *leggibilità*.
- Aumenta la *potenza*. Consente alla funzione di prendere in ingresso *anche* oggetti costanti, per esempio costanti letterali ("pippo").

```
void fun1(string& s) { ... }
```

```
int main(){  
    fun1("pippo");  
    return 0;  
}
```



errore
classico

Prendere puntatori o riferimenti a costante in ingresso permette di:

- 1) Rendere il codice più leggibile. Chi usa la funzione è sicuro che l'argomento in ingresso non verrà modificato.
- 2) Accettare in ingresso anche oggetti costanti, per esempio costanti letterali ("pippo").

Argomento costante in uscita

```
const string& fun1() { ... }  
const char* fun2() { ... }  
const Node* fun3() { ... }
```

- Quando si restituisce un puntatore o riferimento ad un *oggetto gestito internamente*, che non deve essere modificato dall'esterno.
- *Maggiore leggibilità*: il programmatore che usa la funzione sa che non deve modificare l'oggetto.

Restituire in uscita puntatori o riferimenti ad oggetti costanti è più raro. Di solito si usa quando si vuole restituire un puntatore/riferimento ad un oggetto gestito internamente alla funzione/classe, che non deve essere modificato dall'esterno.

Il programmatore che riceve il puntatore/riferimento capisce che non deve modificarlo (più leggibilità).