

Default: `class C {...};`

`C()` → default-constructor      Achtung! Ressourcen

im Zweifelsfall:

**selbst**      `C() {...}; ~C{...}`  
                 `C(const C &init) {...} // copy-constructor`

```
int main (int nargs, char **args)
{
    C einc; // default-constructor oder parameterloser constructor
    C einanderesc = einc; // Initialisierung => copy-constructor
    C eindrittesc; // parameterloser constructor
    eindrittesc = einc; // operator "=", keine Initialisierung

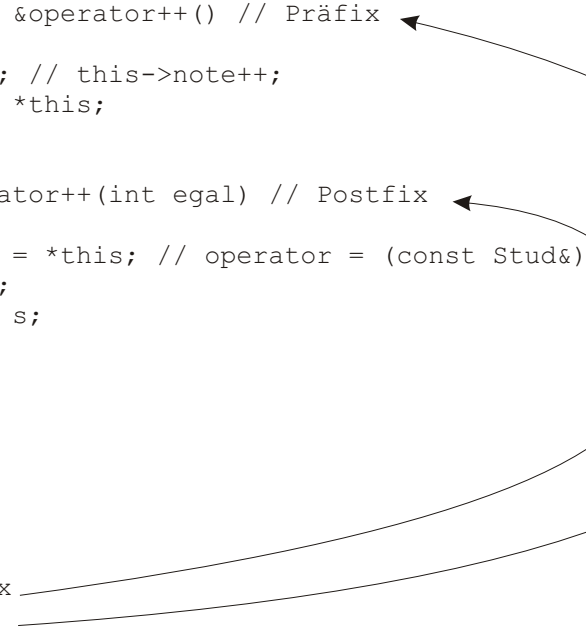
    C *eindrittesc_p = new C(einc); // copy-constructor

    delete eindrittesc_p;

    std::ostream &operator << (std::ostream &s, const c &c);
    std::istream &operator >> (std::istreams, c &c);
    c einc;
    std::cin >> einc;
}
```

```
class Stud
{
    const Stud &operator++() // Präfix ←
    {
        note++; // this->note++;
        return *this;
    }

    Stud &operator++(int egal) // Postfix ←
    {
        Stud s = *this; // operator = (const Stud&)
        note++;
        return s;
    }
    ...
private
    int note;
};
...
Stud s;
s++; // Postfix
++s; // Präfix
Stud s2;
s2 = ++s;
```



in Class: C{...};

private nur Methoden dieser Klasse C

protected nur Methoden dieser Klasse C und alle davon abgeleiteten (falls public oder protected abgeleitet) oder in 1. Ableitung (falls private abgeleitet)

public jeder

Referenz auf 0. Basisklasse:

Zeiger auf 0. Basisklasse:

ggfs. Methode() einer abgeleiteten Klasse wird nur aufgerufen, falls Methode() in der Basisklasse virtual ist.

call-by-value: nie Methode() einer Ableitung