




# C++ cvičení


15.10.2018

[faltin@ksi.mff.cuni.cz](mailto:faltin@ksi.mff.cuni.cz)



## i++ vs. ++i (1/3)

1. `for(int i = 0; i < max; ++i) { do_work(); }`
2. `for(int i = 0; i < max; i++) { do_work(); }`



## i++ vs. ++i (2/3)

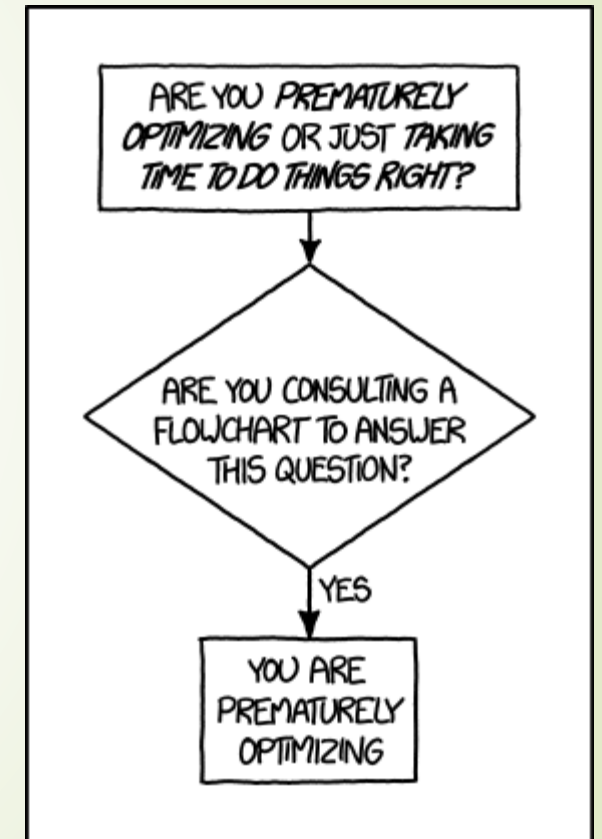
```
T& T::operator++() { // ++i
    ++data;
    return *this;
}
```

```
T T::operator++(int) { // i++
    T copy(*this);
    ++data;
    return copy;
}
```

1. `for(int i = 0; i < max; ++i) { do_work(); }`
2. `for(int i = 0; i < max; i++) { do_work(); }`

# i++ vs. ++i (3/3)

- Premature optimization is the root of all evil!  
-- Knuth
- Benchmark first!



# Deklarace/definice

## **my\_class.hpp**

```
#ifndef MY_CLASS_HPP
#define MY_CLASS_HPP

void fn(int x);

class my_class {
public:
    my_class();
    int exec(int x);

private:
    double d;
};

#endif // MY_CLASS_HPP
```


## **my\_class.cpp**

```
#include "my_class.hpp"
#include <iostream>

void fn(int x) {
    cout << "fn()";
}

my_class::my_class() : d(1.0) {
    cout << "ctor";
}

int my_class::exec(int x) {
    for(int i=0; i < x; ++i) { ... }
}
```



# Kruhová závislost

```
class one; // declaration
class two {
    shared_ptr<one> ptr; // Uses class one.
};

class one : public two {}; // Uses class two.
```



# Úkoly



1. Úkol 3 z minula + třída Complex
  - **poslat mailem do dnešní půlnoci**
2. Piškvorky pro 2 hráče
  - Třídy, OOP, vector<T>

## Úkol 3 (z minula)

```
void fn_copy(C c) {}  
void fn_ref(const C &c) {}
```

```
// Print 1, 2, ..., 20  
int main() {  
    cout << "1";  
    C c;  
    cout << "5";  
    fn_copy(c);  
    cout << "10";  
    fn_ref(c);  
    cout << "15";  
}
```

- Vytvoř třídu/strukturu C (ostatní musí zůstat stejné)