

# Programming in C++

## INTRODUCTION TO C++

- Introduction to basic C++
- Designing professional C++ programs
- Designing with objects and for reuse
- Coding style

## C++ CODING THE PROFESSIONAL WAY

- Classes and objects (Lab 1)
- Inheritance techniques (Lab 2)
- Handling errors and exceptions (Lab 3)
- STL containers, iterators and strings (Lab 4a)
- STL algorithms (Lab 4b)
- I/O Streams (Lab 5)

## MASTERING ADVANCED FEATURES OF C++

- Scope and Namespaces (Lab 6) (*C++ Primer*)
- Operator overloading (Lab 7)
- Polymorphism (Lab 8)
- Templates (Lab 9)
- Extending STL
- Memory management (*valgrind*: Lab 10a)
- Debugging (Lab 10b)
- Profiling (Lab 10c)

## SOME USEFUL C++11 FEATURES

- Multithreaded programming (Lab 11)
- Lambda expressions (Lab 12)
- `auto` (Lab 13a)
- Smart pointer classes (Lab 13b)
- Regular expressions (Lab 14)

## REFERENCES

- Professional C++, *M Gregoire, NA Solter, SJ Kleper* (2011) (text)
- Thinking in C++ (Vol. 1 and 2), *Bruce Eckel*, Pearson (2000) (ref)
- C++ Primer, *SB Lipmann, J Lajoie*, Pearson (2000 or later) (ref)