C++: Presentation

Pierre-Alain Fayolle

Course information

- Lectures:
  - Monday, 4^{th} period, M5
  - Instructor: Pierre-Alain Fayolle, fayolle@u-aizu.ac.jp, room: 323-C
- Exercises:
  - Monday, 5^{th} period, std3 and std4
  - Instructors:
    - std3: Konstantin Markov, markov@u-aizu.ac.jp, room: 126-C
    - std4: Pierre-Alain Fayolle
- In total: 14 lectures, 14 labs and 1 final exam

Course information

- Course objectives: this course provides an introduction to the C++ programming language. In this class, we will:
  - Introduce the students to the basic facilities of the C++ language
  - Present the abstraction mechanisms proposed by the language to allow for: Object Oriented Programming (OOP) and generic programming
  - Give an overview of the standard library and the facilities that it provides (mostly: containers, iterators and algorithms)
- Keywords: C++, Object Oriented Programming (OOP), generic programming, C++ standard library, Standard Template Library (STL)

Plan

- Week 1:
  - Separate compilation
  - Streams
- Week 2:
  - Data abstraction; classes
  - Static
- Week 3: pointers and references
- Week 4:
  - Const correctness
  - Definition and declaration
- Week 5:
  - overloading, constructors (regular, default, copy, conversion) and assignment operator
  - destructors; order of construction and destruction
- Week 6: introduction to inheritance; inheritance and access control; inheritance and substitution principle

Plan

- Week 7: inheritance: virtual methods, overriding vs overloading; abstract base classes
- Week 8: introduction to exceptions
- Week 9: operator overloading
- Week 10:
  - Introduction to generic programming
  - Introduction to templates (function and class)
- Week 11: STL containers
- Week 12: STL iterators
- Week 13: Functors
- Week 14: STL algorithms

References: books

- The C++ programming language. Bjarne Stroustrup
- Accelerated C++. A. Koenig and B. Moo
- More advanced:
  - C++ coding standards: 101 rules, guidelines and best practices. H. Sutter and A. Alexandrescu
  - Effective C++ and More Effective C++. S. Meyers
  - Effective STL. S. Meyers
  - Modern C++ design. A. Alexandrescu

References: web pages

- Course web-site:
  http://www.u-aizu.ac.jp/ fayolle/teaching/2012/C++
- C++ reference: http://www.cppreference.com/wiki/start

Evaluation

Evaluation for the class will be made based on the following:

- Final exam: 45 %
- Exercises: 30 %
- Project: 25 %

Course policy

Academic honesty

- Students are expected to act maturely
- Students are responsible for their actions
- Cheating during exercises, projects or exams is strictly forbidden and will result in failure from the course
- Okay:
  - Discuss and exchange ideas with other students
  - Get ideas from books, web-sites
- Forbidden:
  - Share code with other students
  - Copy code from other students
  - Copy code from other sources without proper attribution and without understanding what the code does

Course policy

- Absence to an exam is equivalent to dropping from the course (i.e. no grade)
- Solution to the exercises should be sent by the students to their instructor and TA by email before the start of the next exercise class
- Exercises submitted late will not be accepted unless the student presents a proper justification to the instructor