IDK1531 Advanced C++ Course

Course Introduction

Aleksandr Lenin

January 29th, 2019

What is this course?

- An introduction to the C++ programming language focusing on syntax and semantics of the language itself, as well as features is offers to the developers.
- Intended for developers who wish to enhance their existing programming experience with a new language.
- Tries to comply with the current C++ standards as much as possible.

By the end of this course the students are able to apply their knowledge of the C++ language coupled with their existing programming skills to:

- Read and understand C++ source code written by other developers.
- Have basic knowledge of the Standard C++ Library.
- Have basic knowledge about some 3rd party libraries.
- Design and create programs that make use of the powerful features that C++ language offers.

Expectations:

- A student knows basic principles of OOP
- A student can design class hierarchy for a particular task
- A student is familiar with data structures: array, set, list, tree, stack, heap, map, queue, …
- A student is familiar with basic algorithms: search and sort, tree traversal, …
- Ability to translate abstract description into code

What this course is not?

- Is not an indtroductory programming course.
- Is not a SW architecture course.
- Is not an OOP design course.

In this course we will cover the following topics:

1. C++ language fundamentals

   entities, types, statements, expressions, literals, identifiers, names, size, alignment, storage duration and its linkage, scope, lifetime, value type, implicit and explicit type conversions, name lookup semantics

2. Pointers and references. Memory handling
3. Exception handling
4. Functions, pointers to functions, function templates, lambda expressions. Value and reference semantics. Lvalue (copy) and rvalue (move) semantics
5. Object initialization, operator overloading, functional objects, pointer to member function, class templates and template specialization

...as well as some parts of the C++ Standard Library.

1. Standard Template Library (STL):
   - Containers Library
   - Iterators Library
   - Algorithms Library
2. Regular Expressions Library
3. Atomic Operations Library
4. Thread Support Library
5. Filesystem Library

+ some external 3rd party libraries.

5 ECTS credits upon completion
1 lecture and 1 practice per week

Schedule:

Lecture:	Tue 12:00–13:30 @U06A–229
Practice:	Wed 16:00 - 17:30 @ICT-121,ICT-122 groups IAPB61,IAPB62 Thu 10:00 - 11:30 @ICT-121, ICT-122 groups IAPB63, IAPB64

Aleksandr Lenin aleksandr.lenin@ttu.ee

Course homepage: https://courses.cs.ttu.ee/pages/IDK1531

The course structured into 3 activity categories:

- Individual assignments – up to 20% of the final grade.
- Course project – 40% of the final grade.
- Tests – up to 40% of the final grade.

Test assessments

- A test work is a pen and paper type assessment
- The knowledge of the C++ language syntax and its features are assessed
- Question types:
  - multiple choice questions
  - specific answers to questions
  - ”fix a problem” type questions
  - mark down correct / incorrect statements

Suggested Reading

- S. Lipmann, J. Lajoie, B. Moo: The C++ Primer. 5th Edition
- Deitel & Deitel: C++ How to Program. 3rd Edition
- P. Anderson and G.Anderson: Navigating C++ and Object-Oriented Design
- S.Meyers: Effective C++. 3rd Edition
- A. Koenig, B. Moo: Accelerated C++. Practical Programming by Example

THANK YOU FOR YOUR ATTENTION

ANY QUESTIONS?