Advanced Programming in C++

Programme course

6 credits

Avancerad programmering i C++

TDDD38

Valid from: 2017 Spring semester

Determined by
Board of Studies for Computer Science
and Media Technology

Date determined
2017-01-25

Main field of study
Computer Science and Engineering, Computer Science, Programming

Course level
Second cycle

Advancement level
A1X

Course offered for
- Computer Engineering, B Sc in Engineering
- Programming
- Computer Science and Engineering, M Sc in Engineering
- Industrial Engineering and Management - International, M Sc in Engineering
- Industrial Engineering and Management, M Sc in Engineering
- Information Technology, M Sc in Engineering
- Computer Science and Software Engineering, M Sc in Engineering
- Applied Physics and Electrical Engineering - International, M Sc in Engineering
- Applied Physics and Electrical Engineering, M Sc in Engineering
- Computer Science, Master's programme
- Communication Systems, Master's programme

Entry requirements
Note: Admission requirements for non-programme students usually also include admission requirements for the programme and threshold requirements for progression within the programme, or corresponding.

Prerequisites
Good knowledge and skills in programming in at least one procedural and/or object-oriented language (e.g. Ada, C, Pascal, Java or C++), and knowledge about fundamentals of object-oriented programming (class, inheritance, polymorhism).

Intended learning outcomes

The purpose of this course is to give deep knowledge about constructs and mechanisms in the programming language C++. Less focus is on advanced applications, i.e., the course is not a systems development course, a problem solving course, or alike. Upon completion of this course the student should be able to:

- Explain non-trivial C++ language constructs and their semantics, e.g. classes, derivation, polymorphism, templates, exception handling, namespaces, types, type conversion, temporary objects, etc.
- Explain the overall design principle of the C++ standard library, especially the container, iterator and algorithms related parts.
- Design and implement usable, correct, error-safe, non-trivial classes and polymorphic classes.
- Design and implement advanced program components, such as e.g. traits classes, policy classes and function object classes.
- Use different components from the C++ standard library in combination to solve non-trivial computation problems, e.g. combine standard containers algorithms, iterators, function objects, an own function objects, to design computations.

Course content

Classes, operator overloading, derived classes, inheritance, polymorphism, lambda expression, namespaces, exception handling, templates. C++ Standard library: strings, streams, containers, iterators, algorithms, function objects, traits and policy classes. C++-specific design patterns.

Teaching and working methods

The course is to a great extent a self-study course, and is given continuously each semester. A series of lectures is given during the first half of the course. Self-studies exercises and different kind of information sources are available on the course web pages. Tutoring is mainly given via e-mail. Computer resourses are scheduled for self-studies. The course is examined four times per year.
The course is offerd twice during 2014: Vt1+Vt2 and Ht1+Ht2.

Examination

DAT1	Computer examination	6 credits	U, 3, 4, 5

There are 4 examination occasions per year, in the examination periods "påsk" (Easter), Vt2 (May/June), August, and Ht2 (January). The computer exam length is 5 hours and IDA's Sun computer system is used, which demands familiarity with that system, the C++ programming environment and text exiting facilities available on that system.

Grades
Four-grade scale, LiU, U, 3, 4, 5

Department
Institutionen för datavetenskap

Director of Studies or equivalent
Ahmed Rezine

Examiner
Klas Arvidsson

Course website and other links
http://www.ida.liu.se/~TDDD38/

Education components
Preliminary scheduled hours: 84 h
Recommended self-study hours: 76 h

Course literature

Additional literature

Books

Bjarne Stroustrup, (2013) *The C++ Programming Language* 4/E Addison-Wesley
Stanley B. Lippman, Josée Lajoie, Barbara E. Moo, (2012) *C++ Primer* 5/E Addison-Wesley

Common rules

Regulations (apply to LiU in its entirety)

The university is a government agency whose operations are regulated by legislation and ordinances, which include the Higher Education Act and the Higher Education Ordinance. In addition to legislation and ordinances, operations are subject to several policy documents. The Linköping University rule book collects currently valid decisions of a regulatory nature taken by the university board, the vice-chancellor and faculty/department boards.

LiU's rule book for education at first-cycle and second-cycle levels is available at http://styrdokument.liu.se/Regelsamling/Innehall/Utbildning_pa_grund-_och_avancerad_niva.