



## DIT300, Compiler Construction, 7.5 ECTS Credits

*Advanced Level*

---

### 1. Establishment

The Faculty Board at the IT-university established the course plan at 2006-11-17. This course plan is effective from autumn 2007.

**Educational area:** Technology/Sciences

### 2. Location

The course is a part of the Computer Science Master's programme and an elective course at Göteborg University.

### 3. Knowledge Requirements

The requirement for the course is to have successfully completed two years of an education aimed at a bachelor degree within Computer Science or equivalent. Specifically, the course DIT230 Programming Languages or equivalent is required.

### 4. Learning Outcomes

The aim of the course is to develop an understanding of the whole process of compiler construction, starting from lexical analysis and finishing with machine code generation. As the course laboration, the students build a complete compiler for a simple imperative language. The course will give the students the knowledge and experience needed to

- define the lexical and syntactic structure of a programming language and implement them efficiently by using standard tools;
- define a type system and implement type checking, which both reports errors to the programmer and prepares the program for code generation;
- know the basic principles of run-time organization, parameter passing, and memory management, and implement them as a part of the compiler;
- be familiar with both stack machines and register-based machines;
- design and implement compilation schemes that generate intermediate or assembly code from a source program;
- use data flow analysis to implement register allocation and code optimizations;
- design and implement extensions of the simple imperative language, and know the main issues of compiling functional and object-oriented languages.

### 5. Content

The teaching consists of lectures, exercises, and laborations, as well as individual supervision in connection to the laborations. The laboration is the central part. The laboration task is a complete compiler for a small imperative programming language. The lectures provide knowledge that is needed in the laboration, as well as an overview of the theoretical foundations and possible extensions of the compiler.

### 6. Literature

See separate literature list.

### **7. Examination**

The course is graded by group exercises and written individual exams.

### **8. Marks**

The course is graded with the following marks: Fail, Pass, Pass with Distinction. The course can also, at the students' request, be marked according to ECTS standards.

### **9. Evaluation**

The course is evaluated through meetings both during and after the course between teachers and student representatives. Further, an anonymous questionnaire can be used to ensure written information. The outcome of the evaluations serves to improve the course by indicating which parts could be added, improved, changed or removed.

### **10. Other**

The course is held in English.