



## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

### DIT622 Databases, 7.5 credits

Databaser, 7,5 högskolepoäng

*First Cycle*

---

#### Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2022-11-17 to be valid from 2023-08-28, autumn semester of 2023.

*Field of education:* Science 100%

*Department:* Department of Computer Science and Engineering

#### Position in the educational system

The course is offered within several programmes. It is also a single subject course at the University of Gothenburg.

The course can be part of the following programmes: 1) Mathematical Sciences, Master's Programme (N2MAT), 2) Computer Science, Master's Programme (N2COS), 3) Applied Data Science Master's Programme (N2ADS), 4) Bachelor's Programme in Mathematics (N1MAT) and 5) Computer Science, Bachelor's Programme (N1COS)

#### *Main field of studies*

Data Science

Computer Science

#### *Specialization*

G1F, First cycle, has less than 60 credits in first-cycle course/s as entry requirements

G1F, First cycle, has less than 60 credits in first-cycle course/s as entry requirements

#### Entry requirements

To be eligible for the course the student must have successfully completed university level courses of at least 45 credits, including:

- at least 15 credits in programming, e.g., DIT440, DIT012, DIT953, or equivalent,
- at least 7.5 credits in mathematics or mathematic reasoning, e.g., DIT980, DIT855, or equivalent,
- concepts in logic, sets, functions and relations that could be acquired with, e.g.,

DIT980, DIT851, or equivalent.

### **Learning outcomes**

On successful completion of the course the student will be able to:

#### *Knowledge and understanding*

- explain the semantic meaning of queries using relational algebra

#### *Competence and skills*

- construct an Entity-Relationship diagram for a given domain
- translate an Entity-Relationship diagram into a relational database schema
- apply design theory concepts for relational databases such as functional dependencies and normalization
- retrieve and modify data using a database language for respective task
- design a database interface using constraints, views, triggers and privileges
- implement a relational database schema and related interface using a data definition language
- communicate with a database, through a database interface, from a software application

#### *Judgement and approach*

- evaluate and create different models for a database domain using Entity-Relationship diagrams and relational schemas
- contrast different data models, such as the relational and the semi-structured data models

### **Course content**

The course covers the basic principles of database systems as seen by users, application programmers and database administrators. A laboratory assignment develops these topics as a running example throughout the course. These include programming in SQL, as seen by a user querying or modifying an existing database, by a database designer, and by an application programmer invoking SQL from a host language.

Course contents include:

- entity-relationship modeling
- functional dependencies and normalisation
- database querying and manipulation through SQL
- interfacing to a database from a host language (such as Java/JDBC)
- overview of non-SQL models (such as JSON or XML)

### *Sub-courses*

- 1. Written hall examination** (*Skriftlig salstentamen* ), 4.5 credits  
Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)
- 2. Laboratory work** (*Laboration*), 3 credits  
Grading scale: Pass (G) and Fail (U)

### **Form of teaching**

The course consists of weekly lectures and exercise sessions, as well as supervised lab sessions.

*Language of instruction:* English

### **Assessment**

Written individual exam given in an examination hall and programming assignments. The programming assignments are normally carried out in pairs.

If a student, who has failed the same examined component twice, wishes to change examiner before the next examination, a written application shall be sent to the department responsible for the course and shall be granted unless there are special reasons to the contrary (Chapter 6, Section 22 of Higher Education Ordinance).

In cases where a course has been discontinued or has undergone major changes, the student shall normally be guaranteed at least three examination occasions (including the ordinary examination) during a period of at least one year from the last time the course was given.

### **Grades**

The grading scale comprises: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U).

To pass the course the student needs to pass both the programming assignments and the written exam.

The final grade in the course is based on the grade of the written exam.

### **Course evaluation**

The course is evaluated through meetings both during and after the course between teachers and student representatives. Further, an anonymous questionnaire is 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.

### **Additional information**

The course is a joint course together with Chalmers.

Course literature to be announced the latest 8 weeks prior to the start of the course.

The course replaces the course DIT621, 7.5 credits. The course cannot be included in a degree which contains DIT621. Neither can the course be included in a degree which is based on another degree in which the course DIT621 is included.