Computer Science for Data Science

Computer Science for Data Science	
Offered by	Department of Mathematics and Computer Science
Language	English
Primarily interesting for	Data Science (BSc) and other BSc students interested in the MSc Data Science in Artificial Intelligence
Prerequisites	Programming, Calculus B (or equivalent)
Contact person	Dr. Dirk Fahland (d.fahland@tue.nl)

Content and composition

This package provides courses to students in the Data Science major, who wish to advance their computer science knowledge. The logic and set theory course is about propositional logic and predicate logic, and you learn to become proficient in modeling and reasoning with logical formulas. The data structures course extends the basic concepts provided in the foundations of computing course. Finally, the algorithmic aspects of data analysis course focus on the application of algorithmic techniques in data analysis problems.

This elective package is for students in the bachelor program Data Science and other programs (except the bachelor program Computer Science), who are interested in the master program Data Science and Artificial Intelligence (DS&AI). The courses in this package offer required and highly recommended prior knowledge for enrollment in this master program. Bachelor Data Science students acquire the necessary knowledge to be admissible to Master program DS&AI if they have successfully completed the following packages

- Data Modeling Foundations for Data Science
- Computer Science for Data Science.

Note: admission to DS&AI requires more prior knowledge than listed in these packages, see https://educationguide.tue.nl/programs/graduate-school/masters-programs/master-data-science-and-ai/?L=2 .

Course code	Course name	Level classification
2IT60*	Logic and set theory	1
2IL50	Data structures	2
JBI045**	Algorithmic aspects of data analysis (optional)	3

* 2IT60 may be replaced by 2ITS60
** JBI045 may be replaced by 2ILC0 if that fits an individual study program better, JBI045 is not an entrance requirement for DS&AI but will aid students in getting more proficient in algorithm design for data analysis

Course description

2IT60, Linear algebra and applications

Logical reasoning is an indispensable tool when designing a solution to any complex technical problem. This course discusses the principles of correct logical reasoning. You learn to formulate statements in unambiguous logical language, and to manipulate statements in a structured and logically valid manner. At the end of the course you are able to give simple mathematical proofs, in particular using the technique of mathematical induction.

Computer Science for Data Science

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

2IL50, Data structures

For solving algorithmic problems many aspects need to be mastered: efficient ways of storing and manipulating (large amounts of) data, algorithm design techniques, how to establish that an algorithm is correct, and how to analyze the efficiency of an algorithm. Design techniques: Incremental algorithms, recursion, divide & conquer. Correctness: induction and invariants. Efficiency analysis: O-notation, recurrences. Sorting: MergeSort, InsertionSort, HeapSort, sorting in linear time, lower bounds for sorting. Selection Algorithms. Data structures: abstract data structures, heaps, hashing, search trees (incl. red-black trees), augmenting data structures, union-find. Basic graph algorithms: adjacency list, adjacency matrix, DFS, BFS, topological sort, minimum spanning trees.

JBI045, Algorithmic aspects of data analysis

This course focuses on the application of algorithmic techniques in data analysis problems. We will consider theoretical aspects—one goal is to show students how data analysis problems can be formulated in precise mathematical terms as optimization or decision problems—but the students will also have to implement various algorithms. We will use an important data-analysis task, namely clustering, as a vehicle for this. Clustering is the task of partitioning a set of input elements into groups by their similarity. The course will give an overview of different approaches to this problem, starting with evaluation metrics—how can we define what a good clustering looks like—and with a focus on algorithms for computing a clustering. We explore how certain data structures and algorithmic techniques can be used to gain considerable speedups in certain situations. We will discuss how the curse of dimensionality affects the clustering problem, and discuss dimension-reduction techniques that can be used to mitigate these effects.