Data Structures and Algorithms for CL 3

Course syllabus, winter semester 2026–2027

Course Description

This programming course provides an introduction to algorithms and data structures that are commonly used in computational linguistics, such as string distance measures, data structures for approximate string search, and finite state automata and related algorithms. The course is driven by assignments where participants implement these algorithms and data structures in small, but practical, applications.

The course language is English.

Topics at glance:

- Analysis of algorithms
- Sorting
- Heaps, priority queues
- Graphs
- Hashes, map data structure
- String algorithms
- Finite-state automata
- Parsing

Prerequisites

The students should be fluent in programming, either able to program in Python, or capable of learning by themselves in a short time. Some familiarity with (computational) linguistics is also assumed.

For the ISCL students, the above requirements are covered in modules ISCL-BA-04 and ASW-BA-02.

Recommended literature

Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser (2013) *Data Structures and Algorithms in Python* (electronic version available through university library)

Daniel Jurafsky and James H. Martin (2009) *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Pearson Prentice Hall, second edition.[1]

[1] Chapters from 3rd edition draft are available at `http://web.stanford.edu/~jurafsky/slp3/`.

Course work and evaluation

The coursework is worth 12 ECTS. Your grade will be determined based on a final exam (80 %) and a project (20 %).[2] The assignments will not contribute to your grade. However, completing **all** weekly assignments is **required** to pass the class. Both assignments are to be done individually. Projects will be done in teams.

If you are too ill to take an exam, or submit your project on time, you must inform the instructor via e-mail, *at least 60 minutes* before the beginning of the exam. You also need to present an attest from your doctor before enrolling into the re-examination. A re-examination is only possible for the exam (and not for the project), and only if you fail the course.

[2] The following grade scale will be used to determine your final grade.

Percent	Local	ECTS
> 96	1.0	A
93–96	1.3	A
89–92	1.7	B
85–88	2.0	B
81–84	2.3	C
77–80	2.7	C
73–76	3.0	C
69–72	3.3	D
65–68	3.7	E
60–64	4.0	E
< 60	5.0	F

Online course environment

We will make use of the utilities offered by Moodle, but more importantly we will use git version management system through GitHub classroom environment for distribution and submission of the assignments. You are required to register the course space on Moodle.[3] The information on online lectures will be posted in this course space. *Please also make sure to obtain a GitHub account, and complete the 'beginning of semester survey' on Moodle.*[4]

[3] https://moodle.zdv.uni-tuebingen.de/course/view.php?id=1980.

[4] https://moodle.zdv.uni-tuebingen.de/mod/feedback/view.php?id=122935.

Academic conduct

You are encouraged to discuss your class work with others, do research on the Internet and use other sources (including large language models with code generation capabilities) for knowledge and inspiration. However, unless stated/cited explicitly, all the coursework you submit should be your own work. You are required to cite any source you have used. If you 'borrow' code that is crucial for the solution of an assignment or a project, you will lose points. Not indicating the source of external code is plagiarism.

Plagiarism or any other form of academic misconduct will not be treated lightly.

Practical information

Instructor	Çağrı Çöltekin ⟨ccoltekin@sfs.uni-tuebingen.de⟩
Office hours	Thu 13:00–14:00
Tutors	Sung-Jin Miriam Han
	How Chuen Yu (Kyle)
Time, place	Mon 14:00–16:00, VG 0.02
	Wed 14:00–16:00, VG 0.02
	Fri 14:00–18:00, VG 0.02 (lab)
Course web page	https://iscl-dsacl3.github.io/