Introduction, administrivia

Data Structures and Algorithms for Computational Linguistics III
ISCL-BA-07

Çağrı Çöltekin
/tʃaːrˈɯ tʃœltecˈɪn/
ccoltekin@sfs.uni-tuebingen.de

University of Tübingen
Seminar für Sprachwissenschaft

Winter Semester 2020/21

What is this course about?

- An intermediate-level course on programming
- Algorithms: (good) solutions to programming problems
- Data structures: (efficient) ways to organize/store information

Prerequisites:

- Data Structures and Algorithms for CL I
- Data Structures and Algorithms for CL II

Module: ISCL-BA-07, Advanced Programming

What is in this course?

A bird's eye view

Introductory lectures on

- Some fundamental data structures: arrays, queues, stacks, trees, ...
- Some fundamental algorithms: searching, sorting, pattern matching, graph algorithms
- Analysis of algorithms
- Finite state automata

Why study algorithms?

- It is one of the fundamental topics in computer science: an algorithm is the way you instruct a computer to do things
- Knowing a clever, efficient solution to one problem helps designing good solutions for other, related problems
- Learning basic algorithmic techniques makes you a better programmer
- Designing good algorithms is an intellectual challenge
- The most popular interview questions for programming jobs are about algorithms

Course overview

- Lectures (all online):
  - Monday 14:15-16:15 (lab)
  - Wednesday 14:15-15:15 (lecture)
  - Friday 14:15-15:15 (lecture)
- Tutors:
  - Anna-Katharina Dick
  - Lea Grüner
- Course website: `https://dsacl3-2020.github.io/`

Literature

- *Data Structures and Algorithms in Python*. Goodrich, Tamassia, and Goldwasser (2013)
  - Available through university library (online version):
    `https://ebookcentral.proquest.com/lib/unitueb/detail.action?docID=4946360`
  - Website of the book contains source code, hints, examples:
    `http://bcs.wiley.com/he-bcs/Books?action=index&bcsId=8029&itemId=1118290275`
- *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Jurafsky and Martin (2009)
  - Draft chapters of 3rd editon is available at
    `https://web.stanford.edu/~jurafsky/slp3/`
- Course notes will be provided for some topics

Coursework and evaluation

- Reading material for most lectures
- 6 programming assignments (approximately every two weeks)
  - The best 5 assignments count (as 60 % of your total grade)
- Final (written) exam (40 %)
- Attendance is not required, but you are unlikely to pass without regular attendance

Assignments

- Assignments in Python
- Only online submissions through GitHub
- Up to one week late = half the points, more than one week late = 0 points
- Solutions will be discussed after the late-assignment deadline
- The assignments can be done in pairs (strongly recommended – knowing your classmates, and learning from them, is an important part of the university experience/education)
- This means **working together on all exercises**, not sharing and parts of an assignment and working on them independently
- We will have a match-making mechanism
- See course page for more information

Assignment 0, Assignment 1

- Please complete Assignment 0, if you haven't done it already
- Assignment 0 allows mapping your GitHub username to student ID
- You will not receive the other assignments without completing it
- Assignment 1 will be released on Monday, Nov 16th, deadline Nov 30
- Questions about the assignments are welcome during tutorial sessions, or on GitHub issues
- If you want a random assignment partner for Assignment 1, please edit the file `assignments-match.txt` in the common private repository by Monday 12:00

Topics at a glance

- A recap of what you should already know: arrays, lists, maps, queues, stacks, iteration, recursion, binary search, ...
- Common algorithmic patterns: brute force, greedy, divide and conquer, dynamic programming, ...
- Algorithmic analysis
- Sorting
- Trees
- Priority queues, heaps
- Hashing
- Graphs, graph algorithms
- Pattern matching
- Tries
- Regular expressions and finite state automata
- Finite state transducers

Final remarks

- Please do not be shy, ask your questions during the lectures
- Please take the assignments seriously, learning programming requires practice
- Next session (Friday): a recap of basic data structures and algorithms

Final remarks

- Please do not be shy, ask your questions during the lectures
- Please take the assignments seriously, learning programming requires practice
- Next session (Friday): a recap of basic data structures and algorithms

- Time for your questions.

Acknowledgments, credits, references

- Some of the slides are based on the previous year's course by Corina Dima.


Goodrich, Michael T., Roberto Tamassia, and Michael H. Goldwasser (2013). *Data Structures and Algorithms in Python*. John Wiley & Sons, Incorporated. ISBN: 9781118476734.


Jurafsky, Daniel and James H. Martin (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. second edition. Pearson Prentice Hall. ISBN: 978-0-13-504196-3.