



Algorithms and Data Structures (NWI-IBC027)

Frits Vaandrager

F.Vaandrager@cs.ru.nl

Institute for Computing and Information Sciences

September 5, 2023



Algorithm

'Algorithm' is arguably the single most important concept in our world

Yuval Harari





What is an Algorithm?

An abstract recipe, prescribing a process that might be carried out by a human, by a computer, or by other means.

David Harel

A set of well-defined rules — a recipe, in effect — for solving some computational problem.

Tim Roughgarden

Ottolenghi's Caramelised Garlic Tart





Recipe for Garlic Tart

Caramelized garlic tart

'I think this is the most delicious recipe in the world!' wrote Claudine after trying it out for me. What else can I add? (See picture on page 40.)

Serves 8

375g all-butter puff pastry

3 medium heads of garlic, cloves separated and peeled

1 tbsp olive oil

1 tsp balsamic vinegar

220ml water

¾ tbsp caster sugar

1 tsp chopped rosemary

1 tsp chopped thyme, plus a few whole sprigs to finish

120g soft, creamy goat's cheese (such as Rosary)

120g hard, mature goat's cheese (such as one from Woolsey Cheese)

2 free-range eggs

100ml double cream

100ml crème fraîche

salt and black pepper

Have ready a shallow, loose-bottomed, 28cm fluted tart tin. Roll out the puff pastry into a circle that will line the bottom and sides of the tin, plus a little extra. Line the tin with the pastry. Place a large circle of greaseproof paper on the bottom and fill up with baking beans. Leave to rest in the fridge for about 20 minutes.

Preheat the oven to 180°C/Gas Mark 4. Place the tart case in the oven and bake blind for 20 minutes. Remove the beans and paper, then bake for a further 5–10 minutes, or until the pastry is golden. Set aside. Leave the oven on.

While the tart case is baking, make the caramelized garlic. Put the cloves in a small saucepan and cover with plenty of water. Bring to a simmer and blanch for 3 minutes, then drain well. Dry the saucepan, return the cloves to it and add the olive oil. Fry the garlic cloves on a high heat for 2 minutes. Add the balsamic vinegar and water and bring to the boil, then simmer gently for 10 minutes. Add the sugar, rosemary, chopped thyme and ¼ teaspoon salt. Continue simmering on a medium flame for 10 minutes, or until most of the liquid has evaporated and the garlic cloves are coated in a dark caramel syrup. Set aside.

To assemble the tart, break both types of goat's cheese into pieces and scatter in the pastry case. Spoon the garlic cloves and syrup evenly over the cheese. In a jug whisk together the eggs, creams, ½ teaspoon salt and some black pepper. Pour this custard over the tart filling to fill the gaps, making sure that you can still see the garlic and cheese over the surface.

Reduce the oven temperature to 160°C/Gas Mark 3 and place the tart inside. Bake for 35–45 minutes, or until the tart filling has set and the top is golden brown. Remove from the oven and leave to cool a little. Then take out of tin, trim the pastry edge if needed, lay a few sprigs of thyme on top and serve warm (it reheats well!) with a crisp salad.

Emotions as Algorithms?

When a woman sees a man and thinks, 'Wow! He is gorgeous!' and when a peahen sees a peacock and thinks, 'Jesus! What a tail!' they are doing something similar to the automatic vending machine. As light reflected from the male's body hits their retinas, extremely powerful algorithms honed by millions of years of evolution kick in. Within a few milliseconds the algorithms convert tiny cues in the male's external appearance into reproduction probabilities, and reach the conclusion: 'In all likelihood, this is a very healthy and fertile male, with excellent genes. If I mate with him, my offspring are also likely to enjoy good health and excellent genes.'

Yuval Harari, *Homo Deus*



Algorithms in this Course

In this course, we consider algorithms in the more restrictive sense of Roughgarden. Nevertheless, it is important to realize that concept of an algorithm/recipe is more general.

We consider algorithms that are both *abstract* and *well-defined*, with *clearly specified input/output behavior*.

Algorithms \neq Programs

F**k the Algorithm?!



Can Algorithms Predict Violent Behavior?

 Net binnen Algemeen Economie Sport Media en Cultuur Achterklap Shop



Maandag 28 augustus 2023 | Het laatste nieuws het eerst op NU.nl



Door onze nieuwsredactie

25 aug 2023 om 18:13
Update: 12 uur geleden

623 reacties

Delen

De politie stopt met het gebruik van een algoritme dat voorspelt wie in de toekomst geweld zou kunnen gebruiken. Dat zegt een woordvoerder van de politie na berichtgeving door *Follow the Money*.



Do Algorithms Make Kids Unhappy?



<https://www.nytimes.com/2023/07/18/opinion/big-tech-algorithms-kids-discovery.html?>

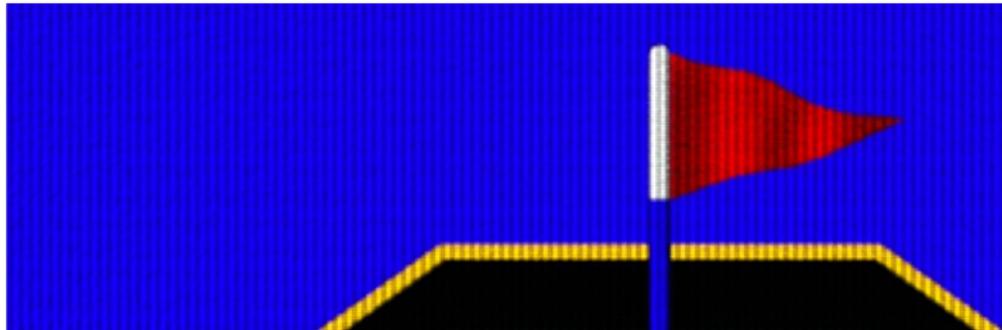
The New York Times

OPINION

GUEST ESSAY

Algorithms Are Making Kids Desperately Unhappy

July 18, 2023



Topic and Objective of this Course

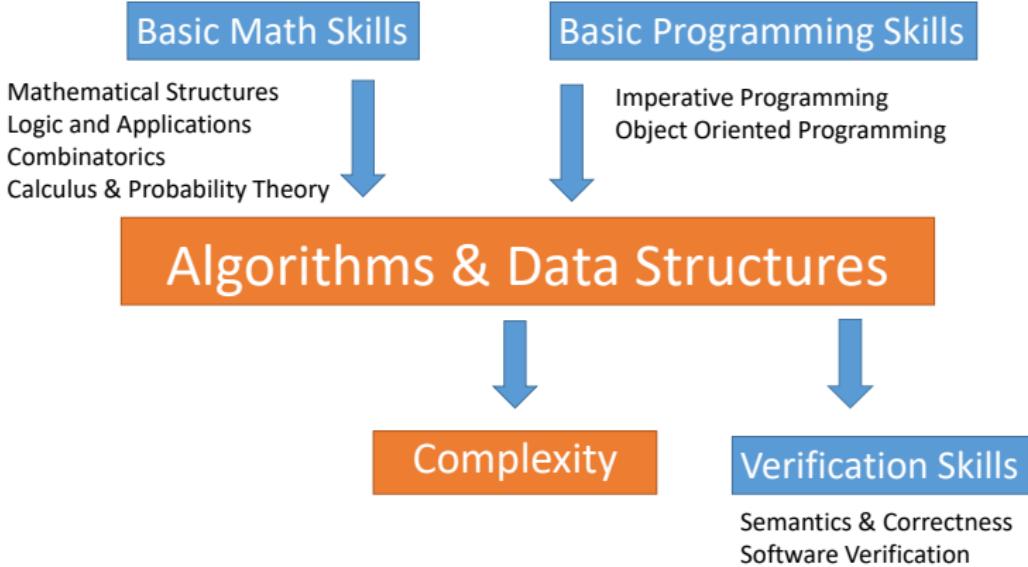
- This course is about *algorithms*, *data structures* and *complexity*
- We *analyze* algorithms and assess their *correctness* and *efficiency*
- Course objective: **to provide the intellectual tools for designing and analyzing algorithms**

Place of Course in the Curriculum



Algorithmics is more than a branch of computer science. It is the core of computer science, and, in all fairness, can be said to be relevant to most of science, business, and technology. David Harel

Place of Course in the Curriculum



ICPC finals 2016 in Thailand



Why Study Algorithms?

- Important for all other branches of computer science
- Driver of technological innovation
- Lens on other sciences
- Useful as a way of thinking
- Fun
- We live in “the age of algorithms”





Course Info

Brightspace

<https://brightspace.ru.nl/>



The Team

Instructors

- Frits Vaandrager
- Niels van der Weide

Teaching assistants (TAs) and PhDs

Sybrand Aarnoutse

Sabin Jacob

Bálint Kocsis

Loes Kruger

Mieke van der Meiden

Tanja Muller

Marnix Suilen

Cas Visser

Thijs van de Griendt

Jochem Kampman

Josja Koopmans

Jasper Laumen

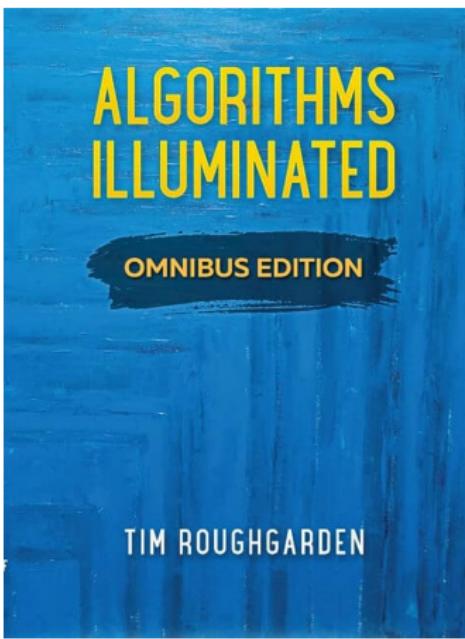
Marijn Meuleman

Fenna Rasing

Niek Terol

Sjeng Vullers

The Book



This textbook is mandatory. Slides, links to tutorials, videos, etc are available through Brightspace.



Activities

Lectures

- Tuesdays, 8.30-10.15
- Active attitude expected: **ask questions!**
- In total there will be 14 lectures
- Recordings of lectures will be made available, as well as links to other relevant videos
- Additional questions may be posed via Discord

Activities

Weekly Problem Sessions

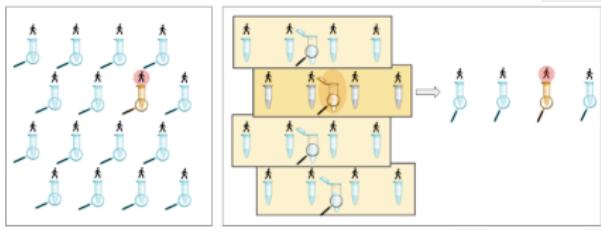
- On campus with groups of ≤ 23 students on Friday 8:30-10:15, Wednesday 15:30-17:15 and Friday 15.30-17.15
- Each student must enroll in a problem session group **today**
- Homework compulsory (9/13)
- Homework problems available by Tuesday 13:30, deadline next Monday at 23:59
- Collaboration is fine but **you must write down your own solution** for the weekly problems!
- **Come to the exercise classes!**

Activities

Two Practical Assignments

- Groups of two (or individually)
- Evaluation with several tests (easy, difficult) using DOMJudge
- You may use Java, C++, C or Python
- Report describing solution, incl. correctness and efficiency
- Average grade students with individual solution usually lower than average grade groups of two
- Prizes for best solutions
- Deadline P1: **17 November** (announced mid October)
- Deadline P2: **22 December** (announced mid November)

Example Practical Assignments





On IP, Plagiarism and the Use of AI Tools

- ① If you copy code snippets from the internet, give proper credit and explain what the code does and why it is correct.
- ② Copying code from the internet or from other students without giving proper credit is **plagiarism** and will be reported to the exam committee.
- ③ If you use tools such as ChatGPT or Copilot, describe precisely how you used them and why you think the produced output is correct.



Activities

Exam

- Written exam + resit on campus

How are Grades Computed?

Grading

- Final grade is $0.6 E + 0.2 P1 + 0.2 P2$, where E is result exam and P1, P2 are results practical assignments
- We require $E \geq 5$
- You may only participate in the exam if you have submitted serious attempts for solutions for both practical assignments and for 9 out of the 13 weekly assignments
- If you did not pass the course previously but your average grade for the practical assignments was ≥ 7 , then you may reuse those grades
- For exam there is a *resit* but not for practical assignments
- You get 0.1 bonus point for exam/*resit* for each weekly assignment with score satisfactory/good, up to 1 bonus point. This bonus point is added to E *before* checking the ≥ 5 requirement.



How Much Time Do I Need?

Time investment (on average)

6ec means $6 \times 28 = 168$ hours in total

- 78 hours for 13 weeks means **6 hours per week** for lectures, studying, problem sessions and small assignments
- 2 hours for the final lecture
- 32 hours for each practical assignment
- 24 hours for preparing exam



How Can I Pass this Course?

How to pass this course ...

- Practice, practice, practice ...
- You don't learn it by just attending lectures and staring at the slides - not a spectator sport!
- Exam questions will be in line with weekly exercises



How Can I Pass this Course?

Last but not least



How Can I Pass this Course?

Last but not least

- You can fail for this course!



How Can I Pass this Course?

Last but not least

- You can fail for this course! Last year around:



How Can I Pass this Course?

Last but not least

- You can fail for this course! Last year around:
- 250 students registered for the course



How Can I Pass this Course?

Last but not least

- You can fail for this course! Last year around:
- 250 students registered for the course
- 175 students were allowed to participate in the exam



How Can I Pass this Course?

Last but not least

- You can fail for this course! Last year around:
- 250 students registered for the course
- 175 students were allowed to participate in the exam
- 110 students passed the course after the first exam

How Can I Pass this Course?

Last but not least

- You can fail for this course! Last year around:
- 250 students registered for the course
- 175 students were allowed to participate in the exam
- 110 students passed the course after the first exam
- 20 students passed the course after the resit

This is an introductory course!

Most algorithms that we will discuss are more than 50 years old.



But algorithms & data structures is an active field, and each year there are major new developments. We will try to make the connection with recent developments.



This week in Amsterdam

The screenshot shows the homepage of the Centrum Wiskunde & Informatica (CWI) website. At the top left is the CWI logo with the text "Research institute for mathematics & computer science in the Netherlands". Below the logo is a navigation bar with links: Research, Education, Results, News (highlighted in red), Events, and Contact. A large image of a modern building with red and green panels and glass windows is visible behind the navigation bar. On the right side of the page, a large watermark of the Radboud University Nijmegen logo is present. The main content area features a red banner with white text: "News < Hundreds of researchers in algorithms gather at CWI" and "Hundreds of researchers in algorithms gather at CWI". Below this, a green box contains the text: "About 300 researchers in algorithms gather together from 4-9 September at the ALGO 2023 conference, which is hosted at CWI."

CWI

Research institute for mathematics & computer science in the Netherlands

Research Education Results News Events Contact

News < Hundreds of researchers in algorithms gather at CWI

Hundreds of researchers in algorithms gather at CWI

About 300 researchers in algorithms gather together from 4-9 September at the ALGO 2023 conference, which is hosted at CWI.

What will we be doing?

- Study examples of beautiful algorithms
- Study examples of powerful data structures
- Discuss correctness and efficiency
- Discuss algorithm design strategies





Topics Addressed in the Course

Lectures

- 1 Introduction and Big-O
- 2 Breadth first search
- 3 Depth first search
- 4 Dijkstra's algorithm
- 5 Flow algorithms 1
- 6 Flow algorithms 2
- 7 Greedy algorithms
- 8 Divide-and-conquer
- 9 Dynamic programming 1
- 10 Dynamic programming 2
- 11 Shortest path revisited
- 12 Search trees
- 13 Hashing
- 14 How to design an algorithm? Question hour



End of this part

Questions?

