Acknowledgement: The course slides are adapted from the slides prepared by R. Sedgewick and K. Wayne of Princeton University.
Introduction

- Why study algorithms?
- Coursework
- Resources
- Outline
Instructor and Course Schedule

• Section I- Erkut ERDEM  
  • erkut@cs.hacettepe.edu.tr

• Section II- Engin Demir  
  • engindemir@cs.hacettepe.edu.tr

• Lectures: Tuesday, 13:00-15:50 @Zoom
• Practicum (BBM204):  
  Wednesday, 16:00-17:50@Zoom  
• Exams: @Gradescope
Instructor and Course Schedule

- Teaching Assistants
  - Nebi Yılmaz  yilmaz@cs.hacettepe.edu.tr
  - Alperen Çakın  alperencakin@cs.hacettepe.edu.tr

- Wednesday, 16:00-15:50@Zoom

- Programming Assignments: @Codepost
- Quizzes: @Gradescope & @Codepost
• This course concerns programming and problem solving, with applications.

• The aim is to teach student how to develop algorithms in order to solve the complex problems in the most efficient way.

• The students are expected to develop a foundational understanding and knowledge of key concepts that underly important algorithms in use on computers today.

• Grading based on
  - Midterm exam 30%
  - Final exam 40%
  - Quizzes (5 out of 6) 30%
• The students are also be expected to gain hand-on experience via a set of programming assignments supplied in the complementary BBM 204 Software Practicum.

• Grading for BBM204 will be based on
  - Participation 4%
  - 4 programming assignments (done individually) 96%
Why study algorithms?

Their impact is broad and far-reaching.

Internet. Web search, packet routing, distributed file sharing, ...

Biology. Human genome project, protein folding, ...

Computers. Circuit layout, file system, compilers, ...

Computer graphics. Movies, video games, virtual reality, ...

Security. Cell phones, e-commerce, voting machines, ...

Multimedia. MP3, JPG, DivX, HDTV, face recognition, ...

Social networks. Recommendations, news feeds, advertisements, ...

Physics. N-body simulation, particle collision simulation, ...
Old roots, new opportunities.

- Study of algorithms dates at least to Euclid.
- Formalized by Church and Turing in 1930s.
- Some important algorithms were discovered by undergraduates in a course like this!
Why study algorithms?

To solve problems that could not otherwise be addressed.

Ex. Network connectivity.
Why study algorithms?

For intellectual stimulation.

“For me, great algorithms are the poetry of computation. Just like verse, they can be terse, allusive, dense, and even mysterious. But once unlocked, they cast a brilliant new light on some aspect of computing.” — Francis Sullivan

“It has often been said that a person does not really understand something until he teaches it to someone else. Actually a person does not really understand something until he can teach it to a computer, i.e. express it as an algorithm The attempt to formalise things as algorithms lead to a much deeper understanding than if we simply try to comprehend things in the traditional way. algorithm must be seen to be believed.” — Donald Knuth
Why study algorithms?

To become a proficient programmer.

“I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important. Bad programmers worry about the code. Good programmers worry about data structures and their relationships.”

— Linus Torvalds (creator of Linux)

“Algorithms + Data Structures = Programs.”

— Niklaus Wirth
Why study algorithms?

They may unlock the secrets of life and of the universe.

Computational models are replacing mathematical models in scientific inquiry.

\[
E = mc^2 \\
F = ma \\
\left[ -\frac{\hbar^2}{2m} \nabla^2 + V(r) \right] \Psi(r) = E \Psi(r)
\]

20\textsuperscript{th} century science
(formula based)

\[
F = \frac{Gm_1m_2}{r^2}
\]

21\textsuperscript{st} century science
(algorithm based)

\[
\text{for (double } t = 0.0; \text{ true; } t = t + dt) \\
\text{ for (int } i = 0; i < N; i++) \\
\text{ \{ \\
\text{ \quad } \text{bodies[i].resetForce();} \\
\text{ \quad } \text{for (int } j = 0; j < N; j++) \\
\text{ \quad \quad } \text{if (i != j) } \\
\text{ \quad \quad \quad } \text{bodies[i].addForce(bodies[j]);} \\
\text{ \} \\
\]

“Algorithms: a common language for nature, human, and computer.” — Avi Wigderson
Why study algorithms?

For fun and profit.
Why study algorithms?

- Their impact is broad and far-reaching.
- Old roots, new opportunities.
- To solve problems that could not otherwise be addressed.
- For intellectual stimulation.
- To become a proficient programmer.
- They may unlock the secrets of life and of the universe.
- For fun and profit.

Why study anything else?
Communication

- The course webpage http://web.cs.hacettepe.edu.tr/~bbm202
- Piazza for discussions and questions. Also it will be updated regularly throughout the semester with lecture notes, programming assignments and important deadlines.

https://piazza.com/hacettepe.edu.tr/spring2021/bbm202
https://piazza.com/hacettepe.edu.tr/spring2021/bbm204
Getting help

- **Office Hours**
- **BBM204 Software Practicum**
  - Course related recitations, practice with algorithms, etc.
- **Communication**
  - Announcements and course related discussions
  - through [piazza](https://piazza.com/hacettepe.edu.tr/spring2021/bbm204)
Coursework and grading

Class participation/Attendance
- Contribute to Piazza discussions.
- Attend and participate in lecture.

Midterm exam 30%
- Scheduled for week 9 (will be in class hours)

Final exam 40%
- Scheduled by Registrar.

Quizzes 30%
- Theoretical and practical aspects
- Top 5 grades out of 6 will be considered
• **Attendance to lectures is mandatory.**

• A student who do not attend the lectures more than 4 weeks will fail BBM202 directly with an F1 grade.

• A student will fail BBM204 directly with an F1 grade if he/she;
  • does not submit more than two assignments
Programming assignments (PAs)

- Four assignments throughout the semester.
- Each assignment has a well-defined goal such as solving a specific problem.
- You must work alone on all assignments stated unless otherwise.
- Unless stated otherwise 2 weeks to work on each assignment

Important Dates

- Programming Assignment 1  3 March
- Programming Assignment 2  31 March
- Programming Assignment 3  21 April
- Programming Assignment 4  12 May
What is cheating?

• Sharing code: by copying, retyping, looking at, or supplying a file
• Coaching: helping your friend to write a programming assignment, line by line
• Copying code from previous course or from elsewhere on WWW

What is NOT cheating?

• Explaining how to use systems or tools
• Helping others with high-level design issues

Penalty for cheating:

• A violation of academic integrity, disciplinary action

Detection of cheating:

• We do check
• Our tools for doing this are much better than most cheaters think!

**Booksite.**
- Brief summary of content.
- Download code from book.

https://algs4.cs.princeton.edu/home/
Course outline

Introduction

Analysis of Algorithms
• Computational Complexity
• Decidability

Sorting
• Elementary Sorting Algorithms,
• Mergesort,
• Quicksort,
• Priority Queues and HeapSort

Searching
• Sequential Search
• Binary Search Trees
• Balanced Trees
• Hashing,
• Search Applications
Course outline

Algorithm Design Techniques
• Dynamic Programming
• Greedy Programming

Graphs
• Undirected Graphs,
• Directed Graphs,
• Minimum Spanning Trees,
• Shortest Path

Strings
• String Sorts, Tries,
• Substring Search,
• Regular Expressions,
• Data Compression

Reductions & Intractability