Lecture 1: Module Introduction

COMP526: Efficient Algorithms

Will Rosenbaum
University of Liverpool

Updated: October 3, 2024

Meeting Goals

- Give an overview of module content
- Describe module organization and philosophy
- Describe module assessment
- Answer your questions about the module
- Learn a bit about one another

Welcome to COMP526!

- Your Instructor:
  Will Rosenbaum
  George Holt 2.16B
  w.rosenbaum@liverpool.ac.uk
- Module Website:
  willrosenbaum.com/teaching/2024f-comp-526
  - The authoritative source for module information about COMP526.
- Poll Everywhere: pollev.com/comp526
  - Used for in-class participation and attendance
  - Use U of L credentials to log in
- CampusWire: https://campuswire.com/p/GBB00CD7A
  - Invite code: **4796**
  - Used for announcements and asynchronous discussion (outside of lecture)

Poll Everywhere Question

Have you used used Poll Everywhere or similar polling software in a module/course before?

pollev.com/comp526

A Bit About Your Instructor

Biographical Sketch:

- Born/grew up near Seattle, WA, USA
- Bachelors degree in mathematics from Reed College in Portland, OR
  - winter weekends snowboarding Mt. Hood
- PhD in mathematics at UCLA
  - Spent weekends hiking mountains of Southern California
  - Also learned to brew beer!
- Postdocs at TAU and Max Planck Institute for Informatics (Germany)
  - Ione (daughter) born!
- Asst. Prof. at Amherst College, USA
- Summer 2024 $\longrightarrow$ **Liverpool, at last**

Research Interests: Theory, Algorithms & Complexity, Distributed Computing

Overview: Goals & Content

Module Goals:

- build / enhance your toolbox of algorithmic methods and techniques
  $\Longrightarrow$ focus on practical methods
- enable you to reason about and communicate algorithmic solutions
  $\Longrightarrow$ level of abstraction, proofs, mathematical analysis, vocabulary
- enable you to apply, combine and extend methods

Overview: Goals & Content

Module Goals:

- build / enhance your toolbox of algorithmic methods and techniques
  $\Longrightarrow$ focus on practical methods
- enable you to reason about and communicate algorithmic solutions
  $\Longrightarrow$ level of abstraction, proofs, mathematical analysis, vocabulary
- enable you to apply, combine and extend methods

Units:

1. Module Overview & Proof Techniques
2. Machines & Models
3. Fundamental Data Structures
4. Efficient Sorting
5. String Matching
6. Compression
7. Error-Correcting Codes
8. Parallel Algorithms
9. Text indexing
10. Streaming Algorithms

Why These Topics?

1. Module Overview & Proof Techniques
2. Machines & Models
3. Fundamental Data Structures
4. Efficient Sorting
5. String Matching
6. Compression
7. Error-Correcting Codes
8. Parallel Algorithms
9. Text indexing
10. Streaming Algorithms

- All fundamental *practical* problems that arise throughout computer science and applications.
- Well-studied problems with efficient solutions.
- Topics of ongoing research—improvements still being made!

Why These Topics?

1. Module Overview & Proof Techniques
2. Machines & Models
3. Fundamental Data Structures
4. Efficient Sorting
5. String Matching
6. Compression
7. Error-Correcting Codes
8. Parallel Algorithms
9. Text indexing
10. Streaming Algorithms

- All fundamental *practical* problems that arise throughout computer science and applications.
- Well-studied problems with efficient solutions.
- Topics of ongoing research—improvements still being made!

Overarching Question

How do we process and organize data *efficiently* so that desired information can be found?

COMP526 in Context

How do the course goals align with Computer Science outside COMP526?

- Practical and efficient solutions for practical problems
- Generic strategies for problem solving in computer science
- Analytical tools for evaluating algorithmic solutions

COMP526 in Context

How do the course goals align with Computer Science outside COMP526?

- Practical and efficient solutions for practical problems
- Generic strategies for problem solving in computer science
- Analytical tools for evaluating algorithmic solutions

When are these skills applicable?

- Writing software
- Evaluating software
- Reading (computer science) literature
- Solving real-world problems
- Nailing technical interviews

Poll Everywhere Question

What are your goals for this module or your reasons for enrolling in the module?

pollev.com/comp526

Module Structure

Course Components, or *"How do we work toward achieving the module goals?"*

1. **Lectures**
    - Meet 2x per week for ~ 90 minutes
    - Primary information dissemination for module
    - Focus on presenting material
    - Readings for before lectures (posted by Friday the week before)
        - Slides will be posted as soon as they're finalized/annotated

Module Structure

Course Components, or *"How do we work toward achieving the module goals?"*

1. **Lectures**
    - Meet 2x per week for ~ 90 minutes
    - Primary information dissemination for module
    - Focus on presenting material
    - Readings for before lectures (posted by Friday the week before)
        - Slides will be posted as soon as they're finalized/annotated
2. **Tutorials**
    - Meet 1x per week for 50 minutes
    - Focus on collaborative problem solving

Module Structure

Course Components, or *"How do we work toward achieving the module goals?"*

1. **Lectures**
   - Meet 2x per week for ~ 90 minutes
   - Primary information dissemination for module
   - Focus on presenting material
   - Readings for before lectures (posted by Friday the week before)
     - Slides will be posted as soon as they're finalized/annotated
2. **Tutorials**
   - Meet 1x per week for 50 minutes
   - Focus on collaborative problem solving
3. **Continuous Assessment** (Coursework)
   - Work you do outside of class
   - Individual practice

Module Structure

Course Components, or *"How do we work toward achieving the module goals?"*

1. **Lectures**
   - Meet 2x per week for ~ 90 minutes
   - Primary information dissemination for module
   - Focus on presenting material
   - Readings for before lectures (posted by Friday the week before)
     - Slides will be posted as soon as they're finalized/annotated
2. **Tutorials**
   - Meet 1x per week for 50 minutes
   - Focus on collaborative problem solving
3. **Continuous Assessment** (Coursework)
   - Work you do outside of class
   - Individual practice
4. **Asynchronous Forum**
   - Discussion mediated through campuswire

Lecture Structure & Rationale

My Goal: for you to learn as much as you can in this module.

- Design Principle: form follows function

A continuum of lecture styles:

Typical lecture (for us):

- Expository lecture material (slides and drawings)
- Problems posed to you for individual/small group consideration
- Short responses/discussion with class as whole
    - mediated through Poll Everywhere

Lecture Structure & Rationale

My Goal: for you to learn as much as you can in this module.

- Design Principle: form follows function

A continuum of lecture styles:

Typical lecture (for us):

- Expository lecture material (slides and drawings)
- Problems posed to you for individual/small group consideration
- Short responses/discussion with class as whole
  - mediated through Poll Everywhere

The Expectation You come to lecture ready to engage and learn.

Why Not a Traditional Lecture?

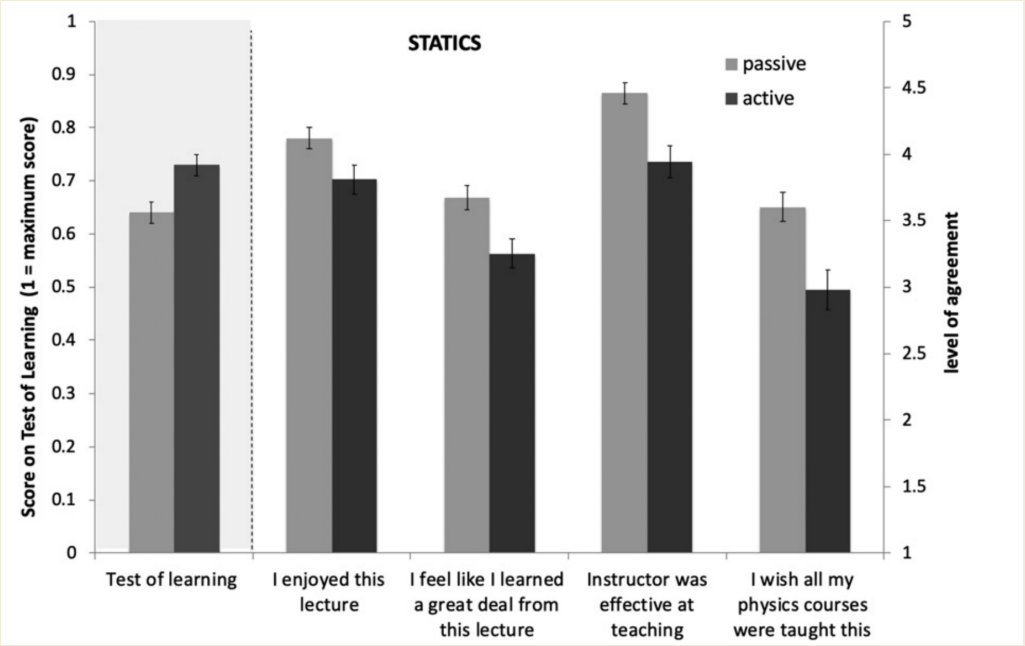
Studies show students in active learning classrooms find:[a]

- sessions are disjointed and lacking flow
- there are frequent interruptions to work
- concerns your errors won't be corrected
- general feeling of frustration and confusion

Why Not a Traditional Lecture?

Studies show students in active learning classrooms find:[a]

- sessions are disjointed and lacking flow
- there are frequent interruptions to work
- concerns your errors won't be corrected
- general feeling of frustration and confusion



[W]hen students experienced confusion and increased cognitive effort with active learning, they perceived this disfluency as a signal of poor learning, while in fact the opposite is true.

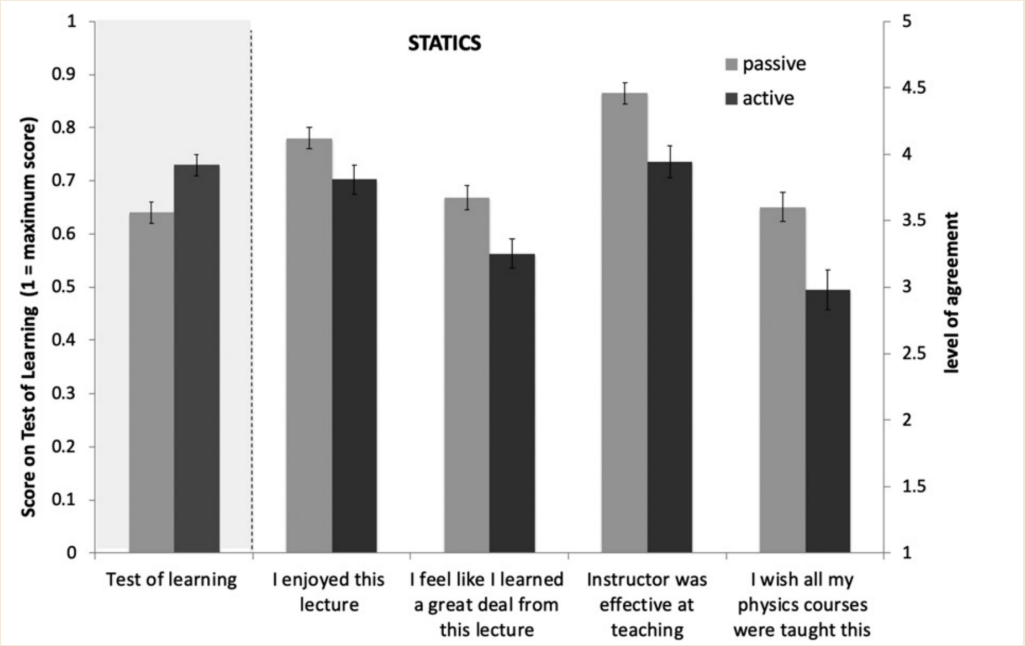
– Deslauriers et al. 2019

[a] See, e.g., Deslauriers et al. 2019, *Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom*

Why Not a Traditional Lecture?

Studies show students in active learning classrooms find:[a]

- sessions are disjointed and lacking flow
- there are frequent interruptions to work
- concerns your errors won't be corrected
- general feeling of frustration and confusion



[W]hen students experienced confusion and increased cognitive effort with active learning, they perceived this disfluency as a signal of poor learning, while in fact the opposite is true.

– Deslauriers et al. 2019

The Moral: Learning doesn't feel the way most students think it should.

[a] See, e.g., Deslauriers et al. 2019, *Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom*

A Note on Attendance

Regular *in-person participation* is expected…

A Note on Attendance

Regular *in-person participation* is expected. . . But stuff comes up!

- Lecture slides will be posted to the course website when they're available
  - annotated slides will be posted after lecture
- Lectures will be recorded and posted to Canvas
- Up to 3 lectures can be missed without penalty
  - Don't come to lecture if you're sick!
- Further absences will result in a reduction in your participation score for the module (5% of total score)
  - Unless you have prior approval/extenuating circumstance as determined by the Student Experience Team (csstudy@liverpool.ac.uk)

Tutorials

Tutorials/Demonstrations meet weekly for 50 minutes
During the sessions you will...

- Work individually and collaboratively on problems related to lecture
- Work with the aid of experienced demonstrators
- See problems similar to those appearing on quizzes and exams
- Have an opportunity to get real-time feedback on your work/ideas

COMP526 Demonstrators:

Seyed Vahidreza (Vahid) Rohani
S.Rohani@liverpool.ac.uk

Juan Pablo Broude (Broude) Garcia
J.Broude-Garcia@liverpool.ac.uk

Continuous Assessment

1. Programming Assignments 1 & 2 **10% of final grade each**
   - Programming assignments spanning 2–3 weeks
   - "Open ended" optimization problems (no single correct answer)
   - Focus on algorithmic/conceptual insight, *not* technical prowess
   - Code in Python, doesn't assume prior experience

Continuous Assessment

1. Programming Assignments 1 & 2 **10% of final grade each**
   - Programming assignments spanning 2–3 weeks
   - "Open ended" optimization problems (no single correct answer)
   - Focus on algorithmic/conceptual insight, *not* technical prowess
   - Code in Python, doesn't assume prior experience
2. Quizzes **15% of final grade total**
   - Weekly quizzes, administered through Canvas
   - Very short: 1 or 2 questions, max 20 minutes
   - Cover previous week's material

Continuous Assessment

1. Programming Assignments 1 & 2 **10% of final grade each**
   - Programming assignments spanning 2–3 weeks
   - "Open ended" optimization problems (no single correct answer)
   - Focus on algorithmic/conceptual insight, *not* technical prowess
   - Code in Python, doesn't assume prior experience
2. Quizzes **15% of final grade total**
   - Weekly quizzes, administered through Canvas
   - Very short: 1 or 2 questions, max 20 minutes
   - Cover previous week's material
3. Class Participation **5% of final grade**
   - Lecture attendance/participation recorded through Poll Everywhere
   - Can miss up to 3 lectures without affect
   - Further absences will affect this score

Continuous Assessment

1. Programming Assignments 1 & 2 **10% of final grade each**
   - Programming assignments spanning 2–3 weeks
   - "Open ended" optimization problems (no single correct answer)
   - Focus on algorithmic/conceptual insight, *not* technical prowess
   - Code in Python, doesn't assume prior experience
2. Quizzes **15% of final grade total**
   - Weekly quizzes, administered through Canvas
   - Very short: 1 or 2 questions, max 20 minutes
   - Cover previous week's material
3. Class Participation **5% of final grade**
   - Lecture attendance/participation recorded through Poll Everywhere
   - Can miss up to 3 lectures without affect
   - Further absences will affect this score

Overall, continuous assessment accounts for 40% of your final grade in the module.

Asynchronous Forum

Outside of lectures, we will use campuswire for asynchronous communication.
Campuswire should be your **first** place to…

- See module announcements
- Discuss lecture material
- Ask clarifying questions about lectures, tutorials, or continuous assessment
- Pose questions to each other related to the module
- Respond to your peer's questions

I will moderate discussion on campuswire, but I strongly encourage peer interaction.

Module Assessments

"How do we check if the module goals have been achieved?"

Main Assessment (60% of final grade): Final Exam

- Questions span topics covered in the module
- Different question types:
  - check factual knowledge
  - simulate known algorithm (by hand)
  - analyze a provided algorithm
  - devise a solution to a novel problem

Module Assessments

"How do we check if the module goals have been achieved?"

Main Assessment (60% of final grade): Final Exam

- Questions span topics covered in the module
- Different question types:
  - check factual knowledge
  - simulate known algorithm (by hand)
  - analyze a provided algorithm
  - devise a solution to a novel problem

Continuous Assessment (40% of final grade)

- Quizzes and lecture participation provide instant feedback on current understanding
- Programming assignments provide opportunities for open ended exploration and applying more creativity in problem solving
  - still towards some concrete, quantitative goal

Poll Everywhere Question

What is the point of assessment?

pollev.com/comp526

What is Expected for Success

- Focused Work

What is Expected for Success

- Focused Work
- Time

What is Expected for Success

- Focused Work
- Time
- Discussion

What is Expected for Success

- Focused Work
- Time
- Discussion
- Peer support

What is Expected for Success

- Focused Work
- Time
- Discussion
- Peer support

My former colleague: *"If I tell you to run 10 miles, it isn't because I want you to be 10 miles away from me."*

Academic Integrity

The University of Liverpool (and the UK in general) as **strict** rules and policies regarding academic integrity.

Academic Integrity

The University of Liverpool (and the UK in general) as **strict** rules and policies regarding academic integrity.

What behaviors constitute violations?

- Using disallowed resources on quizzes or exams.
- Copying from or providing solutions to other students.
- Use any outside resource (including GenAI) in programming assignments *without proper attribution*.

Academic Integrity

The University of Liverpool (and the UK in general) as **strict** rules and policies regarding academic integrity.

What behaviors constitute violations?

- Using disallowed resources on quizzes or exams.
- Copying from or providing solutions to other students.
- Use any outside resource (including GenAI) in programming assignments *without proper attribution*.

What are consequences of violations?

- Consequences range from losing (partial) credit for an assignment (minor violations) to losing credit for the module (major violation) to termination of studies (serious violation).
- For major/serious violations, fault &consequences are determined by University

Academic Integrity

The University of Liverpool (and the UK in general) as **strict** rules and policies regarding academic integrity.

What behaviors constitute violations?

- Using disallowed resources on quizzes or exams.
- Copying from or providing solutions to other students.
- Use any outside resource (including GenAI) in programming assignments *without proper attribution*.

What are consequences of violations?

- Consequences range from losing (partial) credit for an assignment (minor violations) to losing credit for the module (major violation) to termination of studies (serious violation).
- For major/serious violations, fault &consequences are determined by University

Particular resources/expectations will be indicated on each assignment.

Generative Artificial Intelligence

- Large Language Models (LLMs) like ChatGPT, GitHub CoPilot and media generators (Dall-e and Midjourney) are *amazing*.
  - Helpful and surprisingly versatile for "boilerplate" and prototyping
  - Can help shortcut common problems with design/execution of a project

Generative Artificial Intelligence

- Large Language Models (LLMs) like ChatGPT, GitHub CoPilot and media generators (Dall-e and Midjourney) are *amazing*.
  - Helpful and surprisingly versatile for "boilerplate" and prototyping
  - Can help shortcut common problems with design/execution of a project
- But they come with serious drawbacks:
  - Copyright and data collection practices & user privacy
  - Safety/no correctness guarantees
  - Not (yet) good at *reasoning*

Generative Artificial Intelligence

- Large Language Models (LLMs) like ChatGPT, GitHub CoPilot and media generators (Dall-e and Midjourney) are *amazing*.
  - Helpful and surprisingly versatile for "boilerplate" and prototyping
  - Can help shortcut common problems with design/execution of a project
- But they come with serious drawbacks:
  - Copyright and data collection practices & user privacy
  - Safety/no correctness guarantees
  - Not (yet) good at *reasoning*
- Be wary of GenAI for learning
  - The one who does the work is the one who gets the benefit

Generative Artificial Intelligence

- Large Language Models (LLMs) like ChatGPT, GitHub CoPilot and media generators (Dall-e and Midjourney) are *amazing*.
  - Helpful and surprisingly versatile for "boilerplate" and prototyping
  - Can help shortcut common problems with design/execution of a project
- But they come with serious drawbacks:
  - Copyright and data collection practices & user privacy
  - Safety/no correctness guarantees
  - Not (yet) good at *reasoning*
- Be wary of GenAI for learning
  - The one who does the work is the one who gets the benefit
- GenAI should not be used for any substantive part of this module
  - Writing code or designing algorithms for programming assignments
  - Answering questions on quizzes
  - Any GenAI use without attribution

See also UoL Guidelines on GenAI

A Perspective

> *Most of you, if you were students, wrote essays or something like that as undergraduates, right? Fairly confident to say that nobody's actually kept them? Nobody re-reads them. In fact, the essays you wrote are totally worthless.*
>
> *But the value wasn't in the essay. What's valuable is the effort you had to put in to produce the essay. Now, what AI essays do is they shortcut from the request to the delivery of the finished good and bypass the very part of the journey which is actually valuable—the time and effort you invest in constructing the essay in the first place.*
>
> *(Rory Sutherland, "Are We Too Impatient to Be Intelligent?")*

The Moral. The benefits (from this module) come from *struggling* through the work for this module.

- Don't deprive yourself the opportunity to develop skills by taking shortcuts.

Next Time

Logical prerequisites for the analysis of algorithms:

- Propositional and Predicate Logic
- Proof Techniques
- Mathematical Induction

Before Week 03 tutorials be sure to read the posted notes on logic and induction.

- Readings for Lecture 03 will be posted on Friday.

Questions for me?

Scratch Notes