Introduction

EECS 230
Winter 2019
Road map

- What’s it about?
- Policies
- Academic honesty
- How to get help
- A new plan
What EECS 230 is all about

From the course abstract:
What EECS 230 is all about

From the course abstract:

- “EECS 230 teaches foundational programming skills with an emphasis on professionalism.”
What EECS 230 is all about

From the course abstract:

- “EECS 230 teaches foundational programming skills with an emphasis on professionalism.” — we only get you for a quarter, so we want you to acquire the necessary professional skills now
What EECS 230 is all about

From the course abstract:

- “EECS 230 teaches foundational programming skills with an emphasis on professionalism.” — we only get you for a quarter, so we want you to acquire the necessary professional skills now
- “In order to learn to program, we need a language; our language will be C++, but our focus will be on design and pragmatics, not the language itself.”
What EECS 230 is all about

From the course abstract:

- “EECS 230 teaches foundational programming skills with an emphasis on professionalism.” — we only get you for a quarter, so we want you to acquire the necessary professional skills now
- “In order to learn to program, we need a language; our language will be C++, but our focus will be on design and pragmatics, not the language itself.” — language law is for programming languages enthusiasts (like the instructor), but as programmers we care more about what we can do
What EECS 230 is all about

From the course abstract:

- “EECS 230 teaches foundational programming skills with an emphasis on professionalism.” — we only get you for a quarter, so we want you to acquire the necessary professional skills now
- “In order to learn to program, we need a language; our language will be C++, but our focus will be on design and pragmatics, not the language itself.” — language law is for programming languages enthusiasts (like the instructor), but as programmers we care more about what we can do
- “Topics include….”
Topics

- Language basics
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions

- Structuring data: structs and vectors

- Errors and debugging: for when things go wrong

- Classes and generics: defining your own types that work like built-in ones

- Testing: how we know software works (professionalism!)

- Source control: We’ll use Git to protect our work and collaborate with our partners (professionalism)
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics: defining your own types that work like built-in ones
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics: defining your own types that work like built-in ones
- Testing
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics: defining your own types that work like built-in ones
- Testing: how we know software works (professionalism!)
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics: defining your own types that work like built-in ones
- Testing: how we know software works (professionalism!)
- Source control
Topics

- Language basics: expressions, statements, variables, types, assignment, control structures, functions
- Structuring data: structs and vectors
- Errors and debugging: for when things go wrong
- Classes and generics: defining your own types that work like built-in ones
- Testing: how we know software works (professionalism!)
- Source control: We’ll use Git to protect our work and collaborate with our partners (professionalism)
Policies

- There will be a homework assignment every week
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner

- Two exams
  - Thursday, February 7
  - Tuesday, March 12
  - Each worth 15% of your grade

- Mapping of point totals to letter grades is at instructor's discretion
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted

- Two exams
  - Thursday, February 7
  - Tuesday, March 12
  - Each worth 15% of your grade

Mapping of point totals to letter grades is at instructor's discretion.
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted
  - Counts for 70% of your grade

- Two exams
  - Thursday, February 7
  - Tuesday, March 12
  - Each worth 15% of your grade

Mapping of point totals to letter grades is at instructor’s discretion
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted
  - Counts for 70% of your grade
- Two exams

Two exams:
- Thursday, February 7
- Tuesday, March 12

Mapping of point totals to letter grades is at instructor’s discretion.
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted
  - Counts for 70% of your grade

- Two exams
  - Thursday, February 7
  - Tuesday, March 12
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted
  - Counts for 70% of your grade
- Two exams
  - Thursday, February 7
  - Tuesday, March 12
  - Each worth 15% of your grade
Policies

- There will be a homework assignment every week
  - Most will be completed with an assigned partner
  - Late work will not be accepted
  - Counts for 70% of your grade

- Two exams
  - Thursday, February 7
  - Tuesday, March 12
  - Each worth 15% of your grade

- Mapping of point totals to letter grades is at instructor’s discretion
Academic honesty

In EECS 230, we take cheating very seriously.

Cheating is when you:
▶ Receive help of any kind on an exam (except from an instructor)
▶ Give help of any kind on an exam
▶ Share (give or receive) homework code with anyone who is not your official partner
▶ Obtain code from an outside resource, such as Stack Overflow

Please don’t do these things
▶ If you don’t write code, you won’t learn; struggle is good
▶ All cheating will be reported to the relevant dean for investigation

If unsure about your particular situation, ask the instructor or other course staff
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

Please don't do these things
  - If you don't write code, you won't learn; struggle is good
  - All cheating will be reported to the relevant dean for investigation

If unsure about your particular situation, ask the instructor or other course staff
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

Please don't do these things:
- If you don't write code, you won't learn; struggle is good
- All cheating will be reported to the relevant dean for investigation

If unsure about your particular situation, ask the instructor or other course staff
In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

Please don't do these things:
- If you don't write code, you won't learn; struggle is good
- All cheating will be reported to the relevant dean for investigation
- If unsure about your particular situation, ask the instructor or other course staff
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

Please don’t do these things
- If you don’t write code, you won’t learn; struggle is good
- All cheating will be reported to the relevant dean for investigation

If unsure about your particular situation, ask the instructor or other course staff
Academic honesty

In EECS 230, we take cheating very seriously.

- **Cheating is when you:**
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

Please don’t do these things
- If you don’t write code, you won’t learn; struggle is good
- All cheating will be reported to the relevant dean for investigation
- If unsure about your particular situation, ask the instructor or other course staff
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

- Please don’t do these things
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

- Please don’t do these things
  - If you don’t write code, you won’t learn; struggle is good
Academic honesty

In EECS 230, we take cheating very seriously.

- **Cheating is when you:**
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

- **Please don’t do these things**
  - If you don’t write code, you won’t learn; struggle is good
  - All cheating will be reported to the relevant dean for investigation
Academic honesty

In EECS 230, we take cheating very seriously.

- Cheating is when you:
  - Receive help of any kind on an exam (except from an instructor)
  - Give help of any kind on an exam
  - Share (give or receive) homework code with anyone who is not your official partner
  - Obtain code from an outside resource, such as Stack Overflow

- Please don’t do these things
  - If you don’t write code, you won’t learn; struggle is good
  - All cheating will be reported to the relevant dean for investigation

- If unsure about your particular situation, ask the instructor or other course staff
Getting help

- **In person.** Your course staff has office hours:
  
  Instructor:  Jesse Tov  
  
  Gladly by appointment
Getting help

- **In person.** Your course staff has office hours:
  
  Instructor: Jesse Tov           Gladly by appointment
  TAs: John Nguyen              Mon. 2–4 PM
  Yingliao Wang               Tue. 4–6 PM

Times and locations will be listed on the course webpage:

http://users.eecs.northwestern.edu/~jesse/course/eecs230

Online. Ask questions on Piazza:

https://piazza.com/northwestern/winter2019/eecs230
Getting help

- **In person.** Your course staff has office hours:
  - Instructor: Jesse Tov  
    - Gladly by appointment
  - TAs: John Nguyen  
    - Mon. 2–4 PM
    - Yingliao Wang  
    - Tue. 4–6 PM

Times and locations and will be listed on the course webpage:

http://users.eecs.northwestern.edu/~jesse/course/eecs230/
Getting help

- **In person.** Your course staff has office hours:
  
  Instructor: Jesse Tov  
  TAs: John Nguyen  
  Yingliao Wang  
  Gladly by appointment  
  Mon. 2–4 PM  
  Tue. 4–6 PM  

Times and locations and will be listed on the course web page:

http://users.eecs.northwestern.edu/~jesse/course/eecs230/

- **Online.** Ask questions on Piazza:

  https://piazza.com/northwestern/winter2019/eecs230
Problems with EECS 230

- Ill-conceived:
  - You probably can’t become a good programmer in 10 weeks
  - C++ is a very difficult language, especially for beginners

- Not a departmental priority:
  - Enrollment goes down each year as more majors require the 111/211 sequence
  - I’ve been told not to put effort into it!
Course was fairly straightforward and easy to comprehend.

The class taught at an appropriately challenging level. Professor Tov also goes really fast. He communicates the information clearly and gives good in-depth examples in class. He prioritizes learning to code over just giving straight assignments, so homework regrades allowed me to actually correct myself and my code on my own once I understood my mistakes.

Quick response to any homework questions.

Jesse Tov wasn't as bad as the CTECs have previously said, it's just that this class is a real bore if you have any previous programming experience.
CTECs: The good

- “Course was fairly straightforward and easy to comprehend.”
CTECs: The good

- “Course was fairly straightforward and easy to comprehend.”
- “The class taught at an appropriately challenging level. Professor Tov also goes really fast.”
CTECs: The good

- “Course was fairly straightforward and easy to comprehend.”
- “The class taught at an appropriately challenging level. Professor Tov also goes really fast.”
- “He communicates the information clearly and gives good in-depth examples in class. He prioritizes learning to code over just giving straight assignments, so homework regrades allowed me to actually correct myself and my code on my own once I understood my mistakes.”
CTECs: The good

- “Course was fairly straightforward and easy to comprehend.”
- “The class taught at an appropriately challenging level. Professor Tov also goes really fast.”
- “He communicates the information clearly and gives good in-depth examples in class. He prioritizes learning to code over just giving straight assignments, so homework regrades allowed me to actually correct myself and my code on my own once I understood my mistakes.”
- “Quick response to any homework questions”
CTECs: The good

- “Course was fairly straightforward and easy to comprehend.”
- “The class taught at an appropriately challenging level. Professor Tov also goes really fast.”
- “He communicates the information clearly and gives good in-depth examples in class. He prioritizes learning to code over just giving straight assignments, so homework regrades allowed me to actually correct myself and my code on my own once I understood my mistakes.”
- “Quick response to any homework questions”
- “Jesse Tov wasn’t as bad as the CTECs have previously said, it’s just that this class is a real bore if you have any previous programming experience.”
CTECs: The bad

“Utter garbage”

“(H)is last lecture was literally about how pointless C++ is. Like we went through a whole quarter where we’re supposed to be learning a programming language (tbh we were not taught it so I guess whatever) and then we’re told not to use it? That makes no sense.

“Boring and can’t adequately address the needs of the class.”

“The professor is no help at all. His lectures are not that informative, and they do not prepare you for the homework. The midterm’s pretty easy if you read through the notes and know how to read basic code. I recommend it for people who already have coding experience because it’s not that bad if you do.”
CTECs: The bad

- “Utter garbage”
CTECs: The bad

- “Utter garbage”
- “[H]is last lecture was literally about how pointless C++ is. Like we went through a whole quarter where we’re supposed to be learning a programming language (tbh we were not taught it so I guess whatever) and then we’re told not to use it? That makes no sense.”
CTECs: The bad

- “Utter garbage”
- “[H]is last lecture was literally about how pointless C++ is. Like we went through a whole quarter where we’re supposed to be learning a programming language (tbh we were not taught it so I guess whatever) and then we’re told not to use it? That makes no sense.
- “Boring and can’t adequately address the needs of the class.”
CTECs: The bad

- “Utter garbage”
- “[H]is last lecture was literally about how pointless C++ is. Like we went through a whole quarter where we’re supposed to be learning a programming language (tbh we were not taught it so I guess whatever) and then we’re told not to use it? That makes no sense.
- “Boring and can’t adequately address the needs of the class.”
- “The professor is no help at all. His lectures are not that informative, and they do not prepare you for the homework. The midterm’s pretty easy if you read through the notes and know how to read basic code. I recommend it for people who already have coding experience because it’s not that bad if you do.”
CTECs: The ugly

"He doesn't teach. He's rude. He has contempt for his students. I don't understand why he's an educator since he obviously hates it."

"His Piazza responses are heinous. I cannot tell you how much I hated this class. I didn't have to take it. I took it solely for the joy of learning, but obviously Northwestern and Jesse don't care about that."

"The homeworks were tangentially related to what we were doing in class IF that."

"When he came to class the day he accused almost half the class of cheating (I wasn't one of them), he was so rude and spoke with such disgust about the students many of whom I knew did not cheat. We all modeled our code off of his? How was it not supposed to look similar??"
CTECs: The ugly

“He doesn’t teach. He’s rude. He has contempt for his students. I don’t understand why he’s an educator since he obviously hates it.
CTECs: The ugly

“He doesn’t teach. He’s rude. He has contempt for his students. I don’t understand why he’s an educator since he obviously hates it.

“His Piazza responses are heinous. I cannot tell you how much I hated this class. I didn’t have to take it. I took it solely for the joy of learning, but obviously Northwestern and Jesse don’t care about that.
CTECs: The ugly

“He doesn’t teach. He’s rude. He has contempt for his students. I don’t understand why he’s an educator since he obviously hates it.

“His Piazza responses are heinous. I cannot tell you how much I hated this class. I didn’t have to take it. I took it solely for the joy of learning, but obviously Northwestern and Jesse don’t care about that.

“The homeworks were tangentially related to what we were doing in class IF that.
CTECs: The ugly

“He doesn’t teach. He’s rude. He has contempt for his students. I don’t understand why he’s an educator since he obviously hates it.

“His Piazza responses are heinous. I cannot tell you how much I hated this class. I didn’t have to take it. I took it solely for the joy of learning, but obviously Northwestern and Jesse don’t care about that.

“The homeworks were tangentially related to what we were doing in class IF that.

“When he came to class the day he accused almost half the class of cheating (I wasn’t one of them), he was so rude and spoke with such disgust about the students many of whom I knew did not cheat. We all modeled our code off of his? How was it not supposed to look similar??”
Different course, same quarter

“Jesse Tov is incredibly helpful instructor, and he will take the time to make sure the concepts are clear.”

“A clearly deep knowledge about the subject and ability to answer almost any question one might have on type systems. A passion for the material and patience to explain it over (and over) again when we’re struggling to comprehend.”

“Jesse explains concepts in a clear way and provides good motivation for the problems we are solving.”
Different course, same quarter

- “Jesse Tov is incredibly helpful instructor, and he will take the time to make sure the concepts are clear.”
Different course, same quarter

- “Jesse Tov is incredibly helpful instructor, and he will take the time to make sure the concepts are clear.”
- “A clearly deep knowledge about the subject and ability to answer almost any question one might have on type systems. A passion for the material and patience to explain it over (and over) again when we’re struggling to comprehend.”
Different course, same quarter

- “Jesse Tov is incredibly helpful instructor, and he will take the time to make sure the concepts are clear.”
- “A clearly deep knowledge about the subject and ability to answer almost any question one might have on type systems. A passion for the material and patience to explain it over (and over) again when we’re struggling to comprehend.”
- “Jesse explains concepts in a clear way and provides good motivation for the problems we are solving.”
What’s the difference?

That other course was a small seminar. EECS 230 was designed for 65 students and minimal TA support. Now there are only 18 of you! This presents a big opportunity for us.
What’s the difference?

- That other course was a small seminar
What’s the difference?

- That other course was a small seminar
- EECS 230 was designed for 65 students and minimal TA support
What’s the difference?

- That other course was a small seminar
- EECS 230 was designed for 65 students and minimal TA support
- Now there are only 18 of you!
What’s the difference?

- That other course was a small seminar
- EECS 230 was designed for 65 students and minimal TA support
- Now there are only 18 of you!

This presents a big opportunity for us.
Ideas

- Weekly code reviews
- Domain: game design
- Flexible schedule
- Lectures on demand