BILL, RECORD LECTURE!!!!

BILL RECORD LECTURE!!!



Today: Admin, Intro to Theory of Computation

Admin

・ロト・西ト・ヨト・ヨー うへぐ

Necessary Administrative

Everything in these slides is also on the written syllabus on the course website:

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

Necessary Administrative

Everything in these slides is also on the written syllabus on the course website:

https://www.cs.umd.edu/users/gasarch/COURSES/452/S25/
index.html

Necessary Administrative

Everything in these slides is also on the written syllabus on the course website:

https://www.cs.umd.edu/users/gasarch/COURSES/452/S25/ index.html

- 1. Professor: William Gasarch.
- 2. Time: Tu-Th 11:30AM-12:15PM
- 3. Place: IRB 1116.
- TAed by Javier Marinkovic, Alex Mendelsohn, Leo Paranhos.

Necessary administrative stuff

- ► Course Website: Will post slides, notes, and HW there.
- Elms: will post recording.
- Gradescope: you will submit HW there.
 Important Enroll in gradescope using Entry Code 5KYD3Z.
 Do that RIGHT NOW.
- ► Gradescope: we will grade HW there.
- Regrade requests due within a week of the HW being graded.

Piazza is great for asking questions.

Necessary administrative stuff

- ► Course Website: Will post slides, notes, and HW there.
- Elms: will post recording.
- Gradescope: you will submit HW there.
 Important Enroll in gradescope using Entry Code 5KYD3Z.
 Do that RIGHT NOW.
- ► Gradescope: we will grade HW there.
- Regrade requests due within a week of the HW being graded.
- Piazza is great for asking questions.

IF you are auditing this class for whatever reason- perhaps you are having a hard time getting permission to take it, or perhaps you like the material but don't want to take it, let me know and I will put you on the class email list and invite you to join the Piazza.

|▲□▶▲圖▶▲≣▶▲≣▶ = ● のへで

Ask questions in lecture

◆□▶ ◆□▶ ◆目▶ ◆目▶ 目 のへの

- Ask questions in lecture
- Ask questions on piazza

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

- Ask questions in lecture
- Ask questions on piazza

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

Office hours

- Ask questions in lecture
- Ask questions on piazza
- Office hours

Bill: Tu 12:30-3:15 and Th 12:30-3:15 in IRB 2242. Javier: Wed 2:00PM-5:00PM in AV Williams 4160. Alex: Thu 2:00PM-5:00PM in AV Williams 4160. Leo: Mo 1:00PM-2:00PM in AV Williams 4160.

- Ask questions in lecture
- Ask questions on piazza
- Office hours

Bill: Tu 12:30-3:15 and Th 12:30-3:15 in IRB 2242. Javier: Wed 2:00PM-5:00PM in AV Williams 4160. Alex: Thu 2:00PM-5:00PM in AV Williams 4160. Leo: Mo 1:00PM-2:00PM in AV Williams 4160. For office hour location of ALL CS courses see: http://www.cs.umd.edu/class/resources/cstarooms/ fallspring/

ション ふゆ アメビア メロア しょうくしゃ

- Ask questions in lecture
- Ask questions on piazza
- Office hours

Bill: Tu 12:30-3:15 and Th 12:30-3:15 in IRB 2242. Javier: Wed 2:00PM-5:00PM in AV Williams 4160. Alex: Thu 2:00PM-5:00PM in AV Williams 4160. Leo: Mo 1:00PM-2:00PM in AV Williams 4160. For office hour location of ALL CS courses see: http://www.cs.umd.edu/class/resources/cstarooms/ fallspring/

ション ふゆ アメビア メロア しょうくしゃ

Email us- put 452 in the subject line. Bill: gasarch@umd.edu Phone (301) 503-3157 Javier: marinkov@umd.edu Alex: amendel2@umd.edu Leo: lvelloso@terpmail.umd.edu

- Ask questions in lecture
- Ask questions on piazza
- Office hours

Bill: Tu 12:30-3:15 and Th 12:30-3:15 in IRB 2242. Javier: Wed 2:00PM-5:00PM in AV Williams 4160. Alex: Thu 2:00PM-5:00PM in AV Williams 4160. Leo: Mo 1:00PM-2:00PM in AV Williams 4160. For office hour location of ALL CS courses see: http://www.cs.umd.edu/class/resources/cstarooms/ fallspring/

Email us- put 452 in the subject line.
 Bill: gasarch@umd.edu Phone (301) 503-3157
 Javier: marinkov@umd.edu
 Alex: amendel2@umd.edu
 Leo: lvelloso@terpmail.umd.edu

► Appointments (possibly on zoom, possibly at night)

|▲□▶▲圖▶▲≣▶▲≣▶ = ● のへで

Discrete math, modular arithmetic, algorithms, misc math.

▲□▶▲圖▶▲圖▶▲圖▶ 圖 のへで

Discrete math, modular arithmetic, algorithms, misc math.

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

Mathematical maturity.

Discrete math, modular arithmetic, algorithms, misc math.

- Mathematical maturity.
- Ability to write short proofs. (This is not a course like MATH410 where the point is RIGOR.)

- Discrete math, modular arithmetic, algorithms, misc math.
- Mathematical maturity.
- Ability to write short proofs. (This is not a course like MATH410 where the point is RIGOR.)
- There will be some SHORT programming project. (This is not a course like CMSC 412 where the project IS the course.)

- * ロ > * 週 > * 注 > * 注 > ・ 注 - の < @

1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)

- 1. Read notes and slides before class. (**Caution:** Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:

3.1 You can ask questions.

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - **3.2** If you miss class and don't watch the video before the next class you could fall far behind.

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - 3.2 If you miss class and don't watch the video before the next class you could fall far behind.

3.3 Reinforcement Learning.

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - 3.2 If you miss class and don't watch the video before the next class you could fall far behind.
 - 3.3 Reinforcement Learning.
 - 3.4 I sometimes use the whiteboard and that might not get into the recording.

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - 3.2 If you miss class and don't watch the video before the next class you could fall far behind.
 - 3.3 Reinforcement Learning.
 - 3.4 I sometimes use the whiteboard and that might not get into the recording.

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

3.5 Recording might not work that day. In Spring 2024:

- 1. Read notes and slides before class. (Caution: Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - 3.2 If you miss class and don't watch the video before the next class you could fall far behind.
 - 3.3 Reinforcement Learning.
 - 3.4 I sometimes use the whiteboard and that might not get into the recording.

3.5 Recording might not work that day. In Spring 2024: Once recording didn't work and it was my fault.

- 1. Read notes and slides before class. (**Caution:** Some of the slides are in progress. They will be labeled as such. You should not read those, they may contain fake news.)
- 2. Ask questions on Piazza and/or bring questions to class.
- 3. This course will be taped so can catch up or review. However, better to come to class because:
 - 3.1 You can ask questions.
 - 3.2 If you miss class and don't watch the video before the next class you could fall far behind.
 - 3.3 Reinforcement Learning.
 - 3.4 I sometimes use the whiteboard and that might not get into the recording.
 - 3.5 Recording might not work that day. In Spring 2024: Once recording didn't work and it was my fault. Twice recording didn't work and it was not my fault.

▲□▶▲圖▶▲≧▶▲≧▶ ≧ 約९0

1. HWs most weeks.



1. HWs most weeks. Due Tues before class begins.



- 1. HWs most weeks. Due Tues before class begins.
- 2. **Dead Cat Policy:** Can submit HW Thu **before** class-no penalty.

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ
- 1. HWs most weeks. Due Tues before class begins.
- Dead Cat Policy: Can submit HW Thu before class-no penalty.
 YOU were given an extension, so NO extensions past that.

- 1. HWs most weeks. Due Tues before class begins.
- Dead Cat Policy: Can submit HW Thu before class-no penalty.
 YOU were given an extension, so NO extensions past that.

3. We track your morality NOT for grade, but for kicks.

- 1. HWs most weeks. Due Tues before class begins.
- Dead Cat Policy: Can submit HW Thu before class-no penalty.
 YOU were given an extension, so NO extensions past that.
- 3. We track your morality NOT for grade, but for kicks.
- 4. Morally Due Tues Feb 11 means can hand in Thu Feb 13.

- 1. HWs most weeks. Due Tues before class begins.
- Dead Cat Policy: Can submit HW Thu before class-no penalty.
 YOU were given an extension, so NO extensions past that.
- 3. We track your morality NOT for grade, but for kicks.
- 4. Morally Due Tues Feb 11 means can hand in Thu Feb 13. BILL- give thoughts on correlation between morality and grades.

Student I submitted it **Thu** at midnight thinking it was due then, and not at 11:00AM!. Can it still count?

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

Student I submitted it Thu at midnight thinking it was due then, and not at 11:00AM!. Can it still count?Bill Oh, so you submitted it Tues at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you appreciate the Dead-Cat Policy!

Student I submitted it Thu at midnight thinking it was due then, and not at 11:00AM!. Can it still count?Bill Oh, so you submitted it Tues at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you appreciate the Dead-Cat Policy!

Student I forgot to hand it in on **Thu at 11:00AM** but I can prove I did it on time because my computer time stamps my work.

Student I submitted it Thu at midnight thinking it was due then, and not at 11:00AM!. Can it still count?Bill Oh, so you submitted it Tues at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you appreciate the Dead-Cat Policy!

Student I forgot to hand it in on **Thu at 11:00AM** but I can prove I did it on time because my computer time stamps my work. **Bill** Oh, so you forgot to hand it in **Tues at 11:00AM**, then realized this, got it in on Thu before 11:00AM. You are telling me that you **appreciate** the Dead-Cat Policy!

Student I submitted it Thu at midnight thinking it was due then, and not at 11:00AM!. Can it still count?Bill Oh, so you submitted it Tues at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you appreciate the Dead-Cat Policy!

Student I forgot to hand it in on Thu at 11:00AM but I can prove I did it on time because my computer time stamps my work. Bill Oh, so you forgot to hand it in Tues at 11:00AM, then realized this, got it in on Thu before 11:00AM. You are telling me that you appreciate the Dead-Cat Policy! I am not sure why you are telling me about time stamps, but, as

the kids say, whatever.

You can use ChatGPT policy but

<□▶ <□▶ < □▶ < □▶ < □▶ < □▶ < □ > ○ < ○

You can use ChatGPT policy but

1. You must tell me that you did.

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

You can use ChatGPT policy but

 You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.

You can use ChatGPT policy but

 You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.

2. You must rewrite and hand it your own work.

You can use ChatGPT policy but

- You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.
- 2. You must rewrite and hand it your own work.
- 3. **Recommend** If you use it, read what it outputs and make sure its correct (it won't be), well written (it won't be), and that you are enlightened (you won't be).

ション ふゆ アメビア メロア しょうくしゃ

You can use ChatGPT policy but

- You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.
- 2. You must rewrite and hand it your own work.
- 3. **Recommend** If you use it, read what it outputs and make sure its correct (it won't be), well written (it won't be), and that you are enlightened (you won't be).

ション ふゆ アメビア メロア しょうくしゃ

4. Recommend Don't use it.

You can use ChatGPT policy but

- You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.
- 2. You must rewrite and hand it your own work.
- 3. **Recommend** If you use it, read what it outputs and make sure its correct (it won't be), well written (it won't be), and that you are enlightened (you won't be).
- 4. Recommend Don't use it.
- 5. **Upshot** If a problem is standard then ChatGPT will prob get it right, but so will you. If a problem is non-standard then ChatGPT will hallucinate.

ション ふゆ アメビア メロア しょうくしゃ

You can use ChatGPT policy but

- You must tell me that you did. This will explain why what you handed in is so bad that no human would ever have written it.
- 2. You must rewrite and hand it your own work.
- 3. **Recommend** If you use it, read what it outputs and make sure its correct (it won't be), well written (it won't be), and that you are enlightened (you won't be).
- 4. Recommend Don't use it.
- 5. **Upshot** If a problem is standard then ChatGPT will prob get it right, but so will you. If a problem is non-standard then ChatGPT will hallucinate.
- 6. BILL- tell class about how it failed on non-reg, Muffins, Grid Colorings.

You can discuss HW with Humans but hand in your own work.

You can discuss HW with Humans but hand in your own work.

1. You must tell me that you did and who you got help from (if its a student in the class).

▲□▶ ▲□▶ ▲目▶ ▲目▶ | 目 | のへの

You can discuss HW with Humans but hand in your own work.

1. You must tell me that you did and who you got help from (if its a student in the class).

This will explain why what you handed in looks like what they handed in.

ション ふゆ アメビア メロア しょうくしゃ

You can discuss HW with Humans but hand in your own work.

- 1. You must tell me that you did and who you got help from (if its a student in the class).
 - This will explain why what you handed in looks like what they handed in.
 - The most common is that you both made some mistake that nobody else made. If we know this we can discuss and enlighten you!

You can discuss HW with Humans but hand in your own work.

- 1. You must tell me that you did and who you got help from (if its a student in the class).
 - This will explain why what you handed in looks like what they handed in.

The most common is that you both made some mistake that nobody else made. If we know this we can discuss and enlighten you!

2. **Recommend** Make sure you understand what you are handing in.

You can discuss HW with Humans but hand in your own work.

- 1. You must tell me that you did and who you got help from (if its a student in the class).
 - This will explain why what you handed in looks like what they handed in.
 - The most common is that you both made some mistake that nobody else made. If we know this we can discuss and enlighten you!

- 2. **Recommend** Make sure you understand what you are handing in.
- 3. If not you will crash and burn on the written exams.

- ロ > - 4 目 > - 4 目 > - 4 目 > - 9 の 0

1. One midterm, One final,



- 1. One midterm, One final,
- 2. One Prog Project. It will be small.

◆□▶ ◆□▶ ◆ 臣▶ ◆ 臣▶ ○臣 ○ のへぐ

- 1. One midterm, One final,
- 2. One Prog Project. It will be small.
- 3. After midterm I will post an opt project. If you get a D in the course then I will grade it. If you do well in it, $D \rightarrow C-$.

- 1. One midterm, One final,
- 2. One Prog Project. It will be small.
- After midterm I will post an opt project. If you get a D in the course then I will grade it. If you do well in it, D → C−.
 BILL- give thoughts on how the project has worked out.

▲□▶ ▲□▶ ▲ 三▶ ▲ 三▶ ▲ 三 ● ● ●

Textbook

Required Text None. Recommended Text None. If you really want a text then buy used (cheap) or borrow:

Textbook

Required Text None. Recommended Text None. If you really want a text then buy used (cheap) or borrow:

Introduction to The Theory of Computation by Michael Sipser

Textbook

Required Text None. Recommended Text None. If you really want a text then buy used (cheap) or borrow:

Introduction to The Theory of Computation by Michael Sipser There will be notes, slides, and recordings of lecture online.

You are INVITED to talk to us

It has been said that some students do not seek help because they do not know they can.

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

It has been said that some students do not seek help because they do not know they can.

You Can!



It has been said that some students do not seek help because they do not know they can.

You Can!

If **for whatever reason** you are falling behind in the class, or are having trouble with the HW, see us in office hours or **you can make an appointment to see us!** Either in person or on zoom.

Elementary Theory of Computation

*ロ * * @ * * ミ * ミ * ・ ミ * の < や
Our Key Question

Given a problem, classify how hard it is.

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

Given a problem, **classify** how hard it is.

This question permeates all branches of mathematics and computer science.

(ロト (個) (E) (E) (E) (E) のへの

- イロト イ理ト イヨト イヨト ヨー のへぐ

▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

1. How hard is it to express $\sqrt{2}$ as a fraction?

1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible!

1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- 2. How hard is it to solve a quintic equation by expressing its roots by a formula?

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- 2. How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible!

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)
- 3. How hard is it to trisect an angle with a straight edge and compass?

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)
- 3. How hard is it to trisect an angle with a straight edge and compass? **Impossible!**

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)
- How hard is it to trisect an angle with a straight edge and compass? Impossible! (Proven by Wantzel in 1837.)

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)
- How hard is it to trisect an angle with a straight edge and compass? Impossible! (Proven by Wantzel in 1837.)
- 4. There was an awareness of computational problems taking an amount of time, but it was not rigorous.

- 1. How hard is it to express $\sqrt{2}$ as a fraction? Impossible! (Proven by Pythagorean school ~ 600 BC.)
- How hard is it to solve a quintic equation by expressing its roots by a formula? Impossible! (Proven ind. by Galois and Abel in the 1800's.)
- How hard is it to trisect an angle with a straight edge and compass? Impossible! (Proven by Wantzel in 1837.)
- There was an awareness of computational problems taking an amount of time, but it was not rigorous.
 Example: Gauss invented the Fast Fourier Transform but never told anyone since he did not think it was that important.

- イロト イロト イヨト イヨト ヨー のへぐ

 Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z.

 Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z.
 (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- There exists reals that do not solve any polynomial over Z, called Transcendental.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- There exists reals that do not solve any polynomial over Z, called Transcendental.

Liouvlle (1844) showed $\sum_{i=1}^{n} 10^{-n!}$ is T.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- There exists reals that do not solve any polynomial over Z, called Transcendental. Liouvlle (1844) showed ∑ⁿ_{i=1} 10^{-n!} is T.

Liouvlle (1844) showed $\sum_{i=1}^{m} 10^{-m}$ is I Hermite (1873) showed *e* is T.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- There exists reals that do not solve any polynomial over Z, called Transcendental.
 Lioundle (1844) shound ∑ⁿ = 10⁻ⁿ; is T

Liouvlle (1844) showed $\sum_{i=1}^{n} 10^{-n!}$ is T. Hermite (1873) showed *e* is T. Cantor (1874) showed that most numbers are T.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- 2. There exists reals that do not solve any polynomial over \mathbb{Z} , called **Transcendental**.

Liouvlle (1844) showed $\sum_{i=1}^{n} 10^{-n!}$ is T. Hermite (1873) showed *e* is T. Cantor (1874) showed that most numbers are T. von Lindemann (1882) showed that π is T.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- 2. There exists reals that do not solve any polynomial over \mathbb{Z} , called **Transcendental**.

Liouvlle (1844) showed $\sum_{i=1}^{n} 10^{-n!}$ is T. Hermite (1873) showed *e* is T. Cantor (1874) showed that most numbers are T. von Lindemann (1882) showed that π is T.

3. There is no set of axioms from which one can derive all the truths of arithmetic.

- Given a real, we want to know how hard it is in terms of the lowest degree polynomial over Z that it is a root of.
 Example: 2^{1/3} is not the root of any quadratic over Z. (Folklore. This requires a proof which is in the notes, it is optional. It requires linear algebra but is not that hard.)
- There exists reals that do not solve any polynomial over Z, called Transcendental.

Liouvlle (1844) showed $\sum_{i=1}^{n} 10^{-n!}$ is T. Hermite (1873) showed *e* is T. Cantor (1874) showed that most numbers are T. von Lindemann (1882) showed that π is T.

3. There is no set of axioms from which one can derive all the truths of arithmetic. (Godel's Incompleteness Theorem, 1933.)

The Age of Computers.



The Age of Computers.

1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.

▲□▶ ▲□▶ ▲目▶ ▲目▶ 三日 - のへで

The Age of Computers.

1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.

▲□▶ ▲□▶ ▲三▶ ▲三▶ 三三 のへぐ

2. Sorting a list: $\Theta(n \log n)$ comps.

The Age of Computers.

1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.

▲ロ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ● ○ ○ ○

- 2. Sorting a list: $\Theta(n \log n)$ comps.
- 3. $\{a^n b^n : n \in \mathbb{N}\}$ is not regular.

The Age of Computers.

1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.

▲ロ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ● ○ ○ ○

- 2. Sorting a list: $\Theta(n \log n)$ comps.
- 3. $\{a^n b^n : n \in \mathbb{N}\}$ is not regular.
- 4. $\{a^n b^n c^n : n \in \mathbb{N}\}$ is not context-free.

The Age of Computers.

- 1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.
- 2. Sorting a list: $\Theta(n \log n)$ comps.
- 3. $\{a^n b^n : n \in \mathbb{N}\}$ is not regular.
- 4. $\{a^n b^n c^n : n \in \mathbb{N}\}$ is not context-free.
- 5. SAT, other problems, prob not in P (Cook-Levin 1971).

The Age of Computers.

- 1. Searching a sorted list of *n* elts: $lg(n) + \Theta(1)$ comps.
- 2. Sorting a list: $\Theta(n \log n)$ comps.
- 3. $\{a^n b^n : n \in \mathbb{N}\}$ is not regular.
- 4. $\{a^n b^n c^n : n \in \mathbb{N}\}$ is not context-free.
- 5. SAT, other problems, prob not in P (Cook-Levin 1971).

ション ふぼう メリン メリン しょうくしゃ

6. HALT is undecidable (Turing, 1950's.)

Regular Languages.



Regular Languages.

1. These are languages that can be recognized in O(1) space, linear time, and only one pass through the input.

Regular Languages.

- 1. These are languages that can be recognized in O(1) space, linear time, and only one pass through the input.
- There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent. (Most of you saw this in CMSC 330, but Math is best learned twice, at least twice.)

Regular Languages.

- 1. These are languages that can be recognized in O(1) space, linear time, and only one pass through the input.
- There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent. (Most of you saw this in CMSC 330, but Math is best learned twice, at least twice.)
- 3. Regular languages have many closure properties. We will prove this.

Regular Languages.

- 1. These are languages that can be recognized in O(1) space, linear time, and only one pass through the input.
- There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent. (Most of you saw this in CMSC 330, but Math is best learned twice, at least twice.)

- 3. Regular languages have many closure properties. We will prove this.
- 4. We will prove some sets are **not regular**.
▲□▶ ▲□▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへぐ

1. Grammars generate sets of strings.

- 1. Grammars generate sets of strings.
- 2. Two kinds we study: Context Free Grammars (CFG) and Context Sensitive Grammars (CSG).

- 1. Grammars generate sets of strings.
- 2. Two kinds we study: Context Free Grammars (CFG) and Context Sensitive Grammars (CSG).

3. We will study grammars briefly, emphasizing:

- 1. Grammars generate sets of strings.
- 2. Two kinds we study: Context Free Grammars (CFG) and Context Sensitive Grammars (CSG).

- 3. We will study grammars briefly, emphasizing:
 - 3.1 Sets CFGs can generate that are not regular.

- 1. Grammars generate sets of strings.
- 2. Two kinds we study: Context Free Grammars (CFG) and Context Sensitive Grammars (CSG).
- 3. We will study grammars briefly, emphasizing:
 - 3.1 Sets CFGs can generate that are not regular.
 - 3.2 Sets CSGs can generate that CFGs cannot generate.

- 1. Grammars generate sets of strings.
- 2. Two kinds we study: Context Free Grammars (CFG) and Context Sensitive Grammars (CSG).
- 3. We will study grammars briefly, emphasizing:
 - 3.1 Sets CFGs can generate that are not regular.
 - 3.2 Sets CSGs can generate that CFGs cannot generate.
 - 3.3 Problems that can be done with a large DFA or NFA, but only need a small CFG.

Polynomial Time and Non-Deterministic Polynomial Time.

(ロト (個) (E) (E) (E) (E) のへの

Polynomial Time and Non-Deterministic Polynomial Time.

 P is the set of sets that can be decided in Polynomial Time. We think of these problems as being easy.

*ロ * * @ * * ミ * ミ * ・ ミ * の < や

Polynomial Time and Non-Deterministic Polynomial Time.

- P is the set of sets that can be decided in Polynomial Time. We think of these problems as being easy.
- 2. NP-complete we define later. We think of these as problems as being **hard**, though this remains unproven. Why?

Polynomial Time and Non-Deterministic Polynomial Time.

- P is the set of sets that can be decided in Polynomial Time. We think of these problems as being easy.
- NP-complete we define later. We think of these as problems as being hard, though this remains unproven. Why? Because Javier, Alex, and Leo have not worked on it yet.

Polynomial Time and Non-Deterministic Polynomial Time.

- P is the set of sets that can be decided in Polynomial Time. We think of these problems as being easy.
- NP-complete we define later. We think of these as problems as being hard, though this remains unproven. Why? Because Javier, Alex, and Leo have not worked on it yet.
- 3. We will show SAT is NP-complete, and using that, many other problems are NP-complete.

Polynomial Time and Non-Deterministic Polynomial Time.

- P is the set of sets that can be decided in Polynomial Time. We think of these problems as being easy.
- NP-complete we define later. We think of these as problems as being hard, though this remains unproven. Why? Because Javier, Alex, and Leo have not worked on it yet.
- 3. We will show SAT is NP-complete, and using that, many other problems are NP-complete.
- 4. P, NP have many closure properties. We will prove this.

▲□▶ ▲□▶ ▲目▶ ▲目▶ 三日 - のへで

Decidable sets and undecidable sets.

Decidable sets and undecidable sets.

1. DEC is the set of sets that can be decided. We don't care how long it takes!

▲ロ ▶ ▲周 ▶ ▲ ヨ ▶ ▲ ヨ ▶ → ヨ → の Q @

Decidable sets and undecidable sets.

1. DEC is the set of sets that can be decided. We don't care how long it takes!

▲ロ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ● ○ ○ ○

2. UNDEC is the set of all sets that are NOT decidable.

Decidable sets and undecidable sets.

- 1. DEC is the set of sets that can be decided. We don't care how long it takes!
- 2. UNDEC is the set of all sets that are NOT decidable.
- 3. We will show HALT is undecidable and, using that, many other problems are undecidable.

▲ロ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ● ○ ○ ○

Decidable sets and undecidable sets.

- 1. DEC is the set of sets that can be decided. We don't care how long it takes!
- 2. UNDEC is the set of all sets that are NOT decidable.
- 3. We will show HALT is undecidable and, using that, many other problems are undecidable.
- 4. DEC and UNDEC have many closure properties. We will prove this.

▲ロ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ▲ □ ▶ ● ○ ○ ○

Decidable sets and undecidable sets.

- 1. DEC is the set of sets that can be decided. We don't care how long it takes!
- 2. UNDEC is the set of all sets that are NOT decidable.
- 3. We will show HALT is undecidable and, using that, many other problems are undecidable.
- 4. DEC and UNDEC have many closure properties. We will prove this.
- 5. We will define problems that are HARDER THAN HALT.

BILL, STOP RECORDING LECTURE!!!!

▲□▶ ▲圖▶ ▲ 臣▶ ▲ 臣▶ ― 臣 … のへで

BILL STOP RECORDING LECTURE!!!