BILL, RECORD LECTURE!!!!

BILL RECORD LECTURE!!!

Welcome to CMSC 452: Elementary Theory of Computation

Today:
Admin,
Intro to Theory of Computation

Admin

Necessary Administrative

Course Webpage:
https://www.cs.umd.edu/users/gasarch/COURSES/452/S21/index.html

Necessary Administrative

Course Webpage:
https://www.cs.umd.edu/users/gasarch/COURSES/452/S21/index.html

Course Zoom Site:
https://umd.zoom.us/my/gasarch

Necessary administrative stuff

- Course Website: Will post slides, recordings, notes, and HW there. Will NOT be using canvas or elms.
- Gradescope: you will **submit HW** there.
- 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.

Office Hours and Contact Information

Prof Gasarch

- gasarch@cs.umd.edu, (301) 503-3157
- OH Tu & Thur 12:30-2:00
  https://umd.zoom.us/my/gasarch

TA Saadiq Shaik

- saadiqks@gmail.com
- OH Mon 10-12 https://umd.zoom.us/j/6670074227

TA Yaelle Goldschlag

- yaelle.goldschlag@gmail.com
- OH Wed 12-1 https://umd.zoom.us/j/5803841177

TA Eric Shen

- eric.shen2000@gmail.com
- OH Wed 4-5 https://umd.zoom.us/j/6670074227

What You Need For This Class

What You Need For This Class

- Discrete math, modular arithmetic, algorithms, misc math.

What You Need For This Class

- Discrete math, modular arithmetic, algorithms, misc math.
- Mathematical maturity.

What You Need For This Class

- 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.)

What You Need For This Class

- 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 one short programming project. (This is not a course like **Operating Systems** where the project IS the course.)

How to Get the Most Out of This Class

How to Get the Most Out of This 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.)

How to Get the Most Out of This 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.

How to Get the Most Out of This 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:

How to Get the Most Out of This 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:
   3.1 You can ask questions.

How to Get the Most Out of This 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:
   - 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.

How to Get the Most Out of This 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:
   - 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 During COVID try to do **normal** things (Fall CMSC456 students told me that).

How to Get the Most Out of This 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:
   - 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 During COVID try to do **normal** things (Fall CMSC456 students told me that).
   - 3.4 Tape might not always work. (Though it always did in 456.)

HWs/exams

- HWs most weeks.

HWs/exams

- HWs most weeks.
- Due Tuesday **before** class begins. But see next item.

HWs/exams

- HWs most weeks.
- Due Tuesday **before** class begins. But see next item.
- **Dead Cat Policy:** Can submit HW Thursday **before** class without penalty.

HWs/exams

- HWs most weeks.
- Due Tuesday **before** class begins. But see next item.
- **Dead Cat Policy:** Can submit HW Thursday **before** class without penalty.
- **WARNING:** YOU have already been given an extension, HW solutions will be posted on Thursday, so NO extensions past that.

HWs/exams

- HWs most weeks.
- Due Tuesday **before** class begins. But see next item.
- **Dead Cat Policy:** Can submit HW Thursday **before** class without penalty.
- **WARNING:** YOU have already been given an extension, HW solutions will be posted on Thursday, so NO extensions past that.
- We will keep track of your lateness NOT for grade, but for recommendation letters.

HW

What you say, what I hear:

HW

What you say, what I hear:

You Say I thought it was due at midnight!

HW

What you say, what I hear:

You Say I thought it was due at midnight!
I hear Oh, so you submitted it MONDAY at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you **appreciate** the Dead-Cat Policy!

HW

What you say, what I hear:

You Say I thought it was due at midnight!
I hear Oh, so you submitted it MONDAY at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you **appreciate** the Dead-Cat Policy!

You Say Oh, I forgot to hand it in on time but I can prove I did it on time because my computer time stamps my work.

HW

What you say, what I hear:

You Say I thought it was due at midnight!
I hear Oh, so you submitted it MONDAY at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you **appreciate** the Dead-Cat Policy!

You Say Oh, I forgot to hand it in on time but I can prove I did it on time because my computer time stamps my work.
I hear Oh, so you forgot to hand it in MONDAY, then realized this, got it in on Wednesday before rec. You are telling me that you **appreciate** the Dead-Cat Policy!

HW

What you say, what I hear:

You Say I thought it was due at midnight!
I hear Oh, so you submitted it MONDAY at midnight, then realized that the Dead-Cat Policy saved you. You are telling me that you **appreciate** the Dead-Cat Policy!

You Say Oh, I forgot to hand it in on time but I can prove I did it on time because my computer time stamps my work.
I hear Oh, so you forgot to hand it in MONDAY, then realized this, got it in on Wednesday before rec. 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.

Textbook

Required Text None.
Recommended Text None.

Textbook

Required Text None.
Recommended Text None.
There will be notes, slides, and recordings of lecture online.

How to contact Prof or TAs

- Email: Please put "452" in subject line.
- Office hours
- Piazza
- We are around A LOT outside of office hours. It's not as though we're going anywhere!

Elementary Theory of Computation

Our Key Question

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

Our Key Question

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

This question permeates all branches of mathematics and computer science.

The Classification Problem in 600BC-1840AD

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

1. How hard is it to express $\sqrt{2}$ as a fraction? **Impossible!**
   (Proven by Pythagorean school $\sim$ 600 BC.)
2. 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 ruler and compass? **Impossible!**
   (Proven by Wantzel in 1837.)

The Classification Problem in 600BC-1840AD

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

The Classification Problem in 600BC-1840AD

1. How hard is it to express $\sqrt{2}$ as a fraction? **Impossible!**
   (Proven by Pythagorean school $\sim$ 600 BC.)
2. 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 ruler 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.
   Example: Gauss invented the Fast Fourier Transform but never told anyone since he did not think it was that important.

The Classification Problem in 1840AD-1940AD

The Classification Problem in 1840AD-1940AD

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

The Classification Problem in 1840AD-1940AD

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

The Classification Problem in 1840AD-1940AD

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

The Classification Problem in 1840AD-1940AD

1. Given a real, we want to know how hard it is in terms of the lowest degree polynomial over $\mathbb{Z}$ that it is a root of.
   Example: $2^{1/3}$ is not the root of any quadratic over $\mathbb{Z}$.
   (**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 $\mathbb{Z}$.
   (Liouvlle 1844, Hermite 1873, Cantor 1874, von Linderman-e and π.)

The Classification Problem in 1840AD-1940AD

1. Given a real, we want to know how hard it is in terms of the lowest degree polynomial over $\mathbb{Z}$ that it is a root of.
   Example: $2^{1/3}$ is not the root of any quadratic over $\mathbb{Z}$.
   (**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 $\mathbb{Z}$.
   (Liouvlle 1844, Hermite 1873, Cantor 1874, von Linderman-e and π.)
3. There is no set of axioms from which one can derive all the truths of arithmetic.

The Classification Problem in 1840AD-1940AD

1. Given a real, we want to know how hard it is in terms of the lowest degree polynomial over $\mathbb{Z}$ that it is a root of.
   Example: $2^{1/3}$ is not the root of any quadratic over $\mathbb{Z}$.
   (**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 $\mathbb{Z}$.
   (Liouvlle 1844, Hermite 1873, Cantor 1874, von Linderman-e and π.)
3. There is no set of axioms from which one can derive all the truths of arithmetic. (Godel's Incompleteness Theorem, 1933.)

The Classification Problem in 1940AD-2021AD

The Age of Computers.

The Classification Problem in 1940AD-2021AD

The Age of Computers.

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

The Classification Problem in 1940AD-2021AD

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.

The Classification Problem in 1940AD-2021AD

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.

The Classification Problem in 1940AD-2021AD

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.

The Classification Problem in 1940AD-2021AD

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 and many other problems are likely to not be in P.

The Classification Problem in 1940AD-2021AD

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 and many other problems are likely to not be in P.
6. HALT is undecidable (Turing, 1950's.)

Part I of this Course: Regular Languages

Regular Languages.

Part I of this Course: 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.

Part I of this Course: 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.
2. There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent.

Part I of this Course: 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.
2. There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent.
3. Regular languages have many closure properties. We will prove this.

Part I of this Course: 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.
2. There are three different definitions of Regular Languages: DFA, NFA, Regex. We will prove them all equivalent.
3. Regular languages have many closure properties. We will prove this.
4. We will prove some sets are **not regular**.

Part II: Context Free and Context Sensitive Grammars

Part II: Context Free and Context Sensitive Grammars

1. Grammars **generate** sets of strings.

Part II: Context Free and Context Sensitive Grammars

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

Part II: Context Free and Context Sensitive Grammars

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:

Part II: Context Free and Context Sensitive Grammars

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.

Part II: Context Free and Context Sensitive Grammars

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.

Part II: Context Free and Context Sensitive Grammars

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.

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

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

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

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

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

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

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

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

Part III of this Course: P and NP

Polynomial Time and Non-Deterministic Polynomial Time.

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

Part IV of this Course: Decidability and Undecidable

Decidable sets and undecidable sets.

Part IV of this Course: Decidability and Undecidable

Decidable sets and undecidable sets.

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

Part IV of this Course: Decidability and Undecidable

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.

Part IV of this Course: Decidability and Undecidable

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.

Part IV of this Course: Decidability and Undecidable

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.

Part IV of this Course: Decidability and Undecidable

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!!!!

BILL STOP RECORDING LECTURE!!!