BILL AND NATHAN RECORD LECTURE!!!!

BILL AND NATHAN RECORD LECTURE!!!

UN-TIMED PART OF FINAL IS TUESDAY May 11 11:00A. NO DEAD CAT

FINAL IS THURSDAY

May 13

8:00PM-10:15PM

FILL OUT COURSE EVALS for ALL YOUR COURSES!!!

Other Topics I Could Have Covered And Might Next Spring

Other Topics I Could Have Covered And Might Next Spring

Exposition by William Gasarch—U of MD

Steps Forward and Backwards

Complexity theory has its roots in recursion theory.

Steps Forward and Backwards

Complexity theory has its roots in recursion theory.

However, over the last 40 years research in complexity theory has drawn less and less on logic and more and more on combinatorics.

Steps Forward and Backwards

Complexity theory has its roots in recursion theory.

However, over the last 40 years research in complexity theory has drawn less and less on logic and more and more on combinatorics.

A Step Forward means a topic that will help modernize the course. Perhaps any result after 1990.

Steps Forward and Backwards

Complexity theory has its roots in recursion theory.

However, over the last 40 years research in complexity theory has drawn less and less on logic and more and more on combinatorics.

A Step Forward means a topic that will help modernize the course. Perhaps any result after 1990.

A Step Backwards means an old topic, we'll say pre-1980. Often Logic or more tied to the actual machine model. This is not necc bad.

Topics on Reg Langs

Exposition by William Gasarch—U of MD

How Reg Langs are Really Used

How Reg Langs are Really Used

1. Pattern Matching

How Reg Langs are Really Used

1. Pattern Matching
2. Perl-Regular, Ruby-Regular, etc.

How Reg Langs are Really Used

1. Pattern Matching
2. Perl-Regular, Ruby-Regular, etc.
3. Using DFA's to model systems

How Reg Langs are Really Used

1. Pattern Matching
2. Perl-Regular, Ruby-Regular, etc.
3. Using DFA's to model systems
4. Alg to minimize DFAs

How Reg Langs are Really Used

1. Pattern Matching
2. Perl-Regular, Ruby-Regular, etc.
3. Using DFA's to model systems
4. Alg to minimize DFAs

Verdict Have not done. Perl-Regular might drive me nuts since it does not have a clean mathematical semantics.

Desc of Reg Expressions

Theorems about lower bounds on lengths of Regular Expressions.

Desc of Reg Expressions

Theorems about lower bounds on lengths of Regular Expressions.
Verdict Would have to learn those theorems, which I want to.
https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html

Desc of Reg Expressions

Theorems about lower bounds on lengths of Regular Expressions.
Verdict Would have to learn those theorems, which I want to.
`https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`
Goes with the **Length of Description** theme I've had this year.

Desc of Reg Expressions

Theorems about lower bounds on lengths of Regular Expressions.
Verdict Would have to learn those theorems, which I want to.
`https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`
Goes with the **Length of Description** theme I've had this year.
Might want a diff theme next year.

Topics on CFL's

Exposition by William Gasarch—U of MD

Applications

Applications

1. PDA's are DFA's with a stack and are use to model compilers.

Applications

1. PDA's are DFA's with a stack and are use to model compilers.
2. Applications of CFG's and PDA's to Compiler design

Applications

1. PDA's are DFA's with a stack and are use to model compilers.
2. Applications of CFG's and PDA's to Compiler design
3. C++ syntax is undecidable

Applications

1. PDA's are DFA's with a stack and are use to model compilers.
2. Applications of CFG's and PDA's to Compiler design
3. C++ syntax is undecidable

Verdict Won't be covering. Too messy. Will mention these aspects more than I did.

Applications

1. PDA's are DFA's with a stack and are use to model compilers.
2. Applications of CFG's and PDA's to Compiler design
3. C++ syntax is undecidable

Verdict Won't be covering. Too messy. Will mention these aspects more than I did.
Kudos to the person who told me that C++ syntax is undecidable. Good to know!

Desc of Langs

All papers on this are here:
`https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

Desc of Langs

All papers on this are here:
`https://`
`//www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

1. Deterministic PDA's which play into length of descriptions.

Desc of Langs

All papers on this are here:
`https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.

Desc of Langs

All papers on this are here:
`https://`
`//www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.
3. There exists languages L with small CFG's and EENORMOUS DFA's. Much worse than exponential.

Desc of Langs

All papers on this are here:
`https://`
`//www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.
3. There exists languages L with small CFG's and EENORMOUS DFA's. Much worse than exponential.
4. There exists languages L with small TM's and EENORMOUS CFG's. Much worse than exponential.

Desc of Langs

All papers on this are here:
https://www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.
3. There exists languages L with small CFG's and EENORMOUS DFA's. Much worse than exponential.
4. There exists languages L with small TM's and EENORMOUS CFG's. Much worse than exponential.

Verdict Would have to learn those theorems, which I want to.

Desc of Langs

All papers on this are here:
https:
//www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.
3. There exists languages L with small CFG's and EENORMOUS DFA's. Much worse than exponential.
4. There exists languages L with small TM's and EENORMOUS CFG's. Much worse than exponential.

Verdict Would have to learn those theorems, which I want to.
Goes with the **Length of Description** theme I've had this year.

Desc of Langs

All papers on this are here:
`https://`
`//www.cs.umd.edu/users/gasarch/TOPICS/desc/desc.html`

1. Deterministic PDA's which play into length of descriptions.
2. $\{ww : |w| = n\}$ requires exp sized Chomsky Normal Form CFG.
3. There exists languages L with small CFG's and EENORMOUS DFA's. Much worse than exponential.
4. There exists languages L with small TM's and EENORMOUS CFG's. Much worse than exponential.

Verdict Would have to learn those theorems, which I want to.
Goes with the **Length of Description** theme I've had this year.
Might want a diff theme next year.

Topics on Complexity Theory

Exposition by William Gasarch—U of MD

More RESPECT

Recall **RESPECT** is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.

More RESPECT

Recall **RESPECT** is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.
Best Comment by a Student Jevons did not have **RESPECT!**

More RESPECT

Recall **RESPECT** is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.
Best Comment by a Student Jevons did not have **RESPECT!**

1. From the Graph Minor Theorem one obtains MANY problems in P. RESPECT

More RESPECT

Recall RESPECT is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.
Best Comment by a Student Jevons did not have **RESPECT!**

1. From the Graph Minor Theorem one obtains MANY problems in P. RESPECT
2. Hall's Matching Theorem leads to a particular SAT-type problem being in P. RESPECT

More RESPECT

Recall RESPECT is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.

Best Comment by a Student Jevons did not have **RESPECT!**

1. From the Graph Minor Theorem one obtains MANY problems in P. RESPECT
2. Hall's Matching Theorem leads to a particular SAT-type problem being in P. RESPECT
3. SAT Solvers- while not in P, do surprisingly well. RESPECT

More RESPECT

Recall RESPECT is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.

Best Comment by a Student Jevons did not have **RESPECT!**

1. From the Graph Minor Theorem one obtains MANY problems in P. RESPECT
2. Hall's Matching Theorem leads to a particular SAT-type problem being in P. RESPECT
3. SAT Solvers- while not in P, do surprisingly well. RESPECT

Verdict I should write a parody of Aretha Franklin's song RESPECT with this theme.

More RESPECT

Recall RESPECT is shorthand for **Lower Bounds are Hard because you never know when someone will come along with clever math or deep math or SOMETHING that your so-called lower bound did not take into account**.

Best Comment by a Student Jevons did not have **RESPECT!**

1. From the Graph Minor Theorem one obtains MANY problems in P. RESPECT
2. Hall's Matching Theorem leads to a particular SAT-type problem being in P. RESPECT
3. SAT Solvers- while not in P, do surprisingly well. RESPECT

Verdict I should write a parody of Aretha Franklin's song RESPECT with this theme.

Also, would be happy to do any of these topics.

SEND+MORE=MONEY

$$\begin{array}{ccccc} & S & E & N & D \\ + & M & O & R & E \\ \hline M & O & N & E & Y \end{array}$$

SEND+MORE=MONEY

$$\begin{array}{ccccc} & S & E & N & D \\ + & M & O & R & E \\ \hline M & O & N & E & Y \end{array}$$

Has Solution

$$\begin{array}{ccccc} & 9 & 5 & 6 & 7 \\ + & 1 & 0 & 8 & 5 \\ \hline 1 & 0 & 6 & 5 & 2 \end{array}$$

SEND+MORE=MONEY

	S	E	N	D
+	M	O	R	E
M	O	N	E	Y

Has Solution

	9	5	6	7
+	1	0	8	5
1	0	6	5	2

Given a puzzle, does it have a solution, is NP-complete

SEND+MORE=MONEY

	S	E	N	D
+	M	O	R	E
M	O	N	E	Y

Has Solution

	9	5	6	7
+	1	0	8	5
1	0	6	5	2

Given a puzzle, does it have a solution, is NP-complete

Verdict Not sure. Good to see one hard reduction. Too hard?

Complexity of Grid Coloring

Def A c-coloring of an $n \times m$ grid is a coloring that has no monochromatic rectangles.

Complexity of Grid Coloring

Def A c-coloring of an $n \times m$ grid is a coloring that has no monochromatic rectangles.

Problem Given a partial c-coloring of an $n \times m$ grid can it be extended to a coloring of the entire grid?

Complexity of Grid Coloring

Def A c-coloring of an $n \times m$ grid is a coloring that has no monochromatic rectangles.

Problem Given a partial c-coloring of an $n \times m$ grid can it be extended to a coloring of the entire grid?

Thm The problem is NP-complete.

Complexity of Grid Coloring

Def A c-coloring of an $n \times m$ grid is a coloring that has no monochromatic rectangles.

Problem Given a partial c-coloring of an $n \times m$ grid can it be extended to a coloring of the entire grid?

Thm The problem is NP-complete.

Verdict Probably to hard and obscure.

Complexity of Grid Coloring

Def A c-coloring of an $n \times m$ grid is a coloring that has no monochromatic rectangles.

Problem Given a partial c-coloring of an $n \times m$ grid can it be extended to a coloring of the entire grid?

Thm The problem is NP-complete.

Verdict Probably to hard and obscure.
I would know– it was my open problem and I am an author on the paper that solved it.

Other Problems that are Hard

Other Problems that are Hard

1. CHESS is EXPTIME-complete

Other Problems that are Hard

1. CHESS is EXPTIME-complete
2. GO is EXPTIME-complete

Other Problems that are Hard

1. CHESS is EXPTIME-complete
2. GO is EXPTIME-complete
3. Equiv of trex is EXPSPACE-complete

Other Problems that are Hard

1. CHESS is EXPTIME-complete
2. GO is EXPTIME-complete
3. Equiv of trex is EXPSPACE-complete

Verdict trex ties into the other parts of the course. But all of these proof are similar to Cook-Levin so messy TM stuff. A Step Backwards.

Bounded Queries in Complexity Theory

$\exists$ fnctns computable in poly time with 5 queries to SAT but not 4?
Assuming $\mathrm{P} \neq \mathrm{NP}$, YES.

Bounded Queries in Complexity Theory

$\exists$ fnctns computable in poly time with 5 queries to SAT but not 4?
Assuming $\mathrm{P} \neq \mathrm{NP}$, YES.

$\exists$ sets decidable in poly time with 5 queries to SAT but not 4?

Bounded Queries in Complexity Theory

$\exists$ fnctns computable in poly time with 5 queries to SAT but not 4?
Assuming $\mathrm{P} \neq \mathrm{NP}$, YES.

$\exists$ sets decidable in poly time with 5 queries to SAT but not 4?
Assuming $\Sigma_2^p \neq \Pi_2^p$, YES.

Bounded Queries in Complexity Theory

$\exists$ fnctns computable in poly time with 5 queries to SAT but not 4?
Assuming $\mathrm{P} \neq \mathrm{NP}$, YES.

$\exists$ sets decidable in poly time with 5 queries to SAT but not 4?
Assuming $\Sigma_2^p \neq \Pi_2^p$, YES.
Verdict Number of queries as a complexity measure is interesting.
Would be happy to do these topics.

Lower Bounds on Approx using PCP Thm

The PCP theorem gives a characterization of NP which is useful for proving lower bounds on approximation.

Lower Bounds on Approx using PCP Thm

The PCP theorem gives a characterization of NP which is useful for proving lower bounds on approximation.

1. Proving PCP Thm takes to long to do in this class. (It was even to long to do in CMSC 858.)

Lower Bounds on Approx using PCP Thm

The PCP theorem gives a characterization of NP which is useful for proving lower bounds on approximation.

1. Proving PCP Thm takes to long to do in this class. (It was even to long to do in CMSC 858.)
2. Using PCP to show that CLIQ is hard to approximate is plausible.

Lower Bounds on Approx using PCP Thm

The PCP theorem gives a characterization of NP which is useful for proving lower bounds on approximation.

1. Proving PCP Thm takes to long to do in this class. (It was even to long to do in CMSC 858.)
2. Using PCP to show that CLIQ is hard to approximate is plausible.
3. Using PCP to show that other problems hard to approximate is plausible.

Lower Bounds on Approx using PCP Thm

The PCP theorem gives a characterization of NP which is useful for proving lower bounds on approximation.

1. Proving PCP Thm takes to long to do in this class. (It was even to long to do in CMSC 858.)
2. Using PCP to show that CLIQ is hard to approximate is plausible.
3. Using PCP to show that other problems hard to approximate is plausible.

Verdict A Step Forward! Might be to hard.

Why we think GI is Not NPC

Thm If GI is NPC then $\Sigma_2^p = \Pi_2^p$.

Why we think GI is Not NPC

Thm If GI is NPC then $\Sigma_2^p = \Pi_2^p$.

1. Involves some probability

Why we think GI is Not NPC

Thm If GI is NPC then $\Sigma_2^p = \Pi_2^p$.

1. Involves some probability
2. Would take 2 or 3 lectures.

Why we think GI is Not NPC

Thm If GI is NPC then $\Sigma_2^p = \Pi_2^p$.

1. Involves some probability
2. Would take 2 or 3 lectures.
3. Is very interesting.

Why we think GI is Not NPC

Thm If GI is NPC then $\Sigma_2^p = \Pi_2^p$.

1. Involves some probability
2. Would take 2 or 3 lectures.
3. Is very interesting.

Verdict A Step Forward! Might be to hard.

Theorems from Complexity Theory

Theorems from Complexity Theory

1. A set A is **sparse** if $\exists$ poly p, $|A \cap \Sigma^n| \leq p(n)$.

Theorems from Complexity Theory

1. A set A is **sparse** if $\exists$ poly p, $|A \cap \Sigma^n| \leq p(n)$.
   Thm If a sparse set is NP-complete then P=NP.

Theorems from Complexity Theory

1. A set A is **sparse** if $\exists$ poly p, $|A \cap \Sigma^n| \leq p(n)$.
   Thm If a sparse set is NP-complete then P=NP.
2. Theorems on the Poly Hierarchy.

Theorems from Complexity Theory

1. A set A is **sparse** if $\exists$ poly p, $|A \cap \Sigma^n| \leq p(n)$.
   Thm If a sparse set is NP-complete then P=NP.
2. Theorems on the Poly Hierarchy.
3. For all time bounds $T(n)$ there is a decidable set NOT in $\mathrm{DTIME}(T(n))$.

Theorems from Complexity Theory

1. A set A is **sparse** if $\exists$ poly p, $|A \cap \Sigma^n| \le p(n)$.
   Thm If a sparse set is NP-complete then P=NP.
2. Theorems on the Poly Hierarchy.
3. For all time bounds $T(n)$ there is a decidable set NOT in $\mathrm{DTIME}(T(n))$.

Verdict I have done both of these in class and may do it again. A tiny step backwards.

Theorems from Space Complexity

1. Nondet-Log-Space is closed under complement.

Theorems from Space Complexity

1. Nondet-Log-Space is closed under complement.
2. Nondet-Log-Space is contained in P.

Theorems from Space Complexity

1. Nondet-Log-Space is closed under complement.
2. Nondet-Log-Space is contained in P.
3. $\mathrm{NSPACE}(S(n)) \subseteq \mathrm{DSPACE}(S(n)^2)$.

Theorems from Space Complexity

1. Nondet-Log-Space is closed under complement.
2. Nondet-Log-Space is contained in P.
3. $\mathrm{NSPACE}(S(n)) \subseteq \mathrm{DSPACE}(S(n)^2)$.

Verdict All nice theorems that I could do. Would need to introduce and talk about space complexity so this would take time. Not that hard, so thats good.

Caveat Space Complexity is not as much fun as a theme as RESPECT is.

Primitive Recursive Functions

Def Prim Rec Functions are in between P and Undecidable.

Primitive Recursive Functions

Def Prim Rec Functions are in between P and Undecidable.

What They Include Exp, double-exp, Tower, WOWER, etc.

Primitive Recursive Functions

Def Prim Rec Functions are in between P and Undecidable.

What They Include Exp, double-exp, Tower, WOWER, etc.

Where Used In some branches of Math Prim-rec vs non-prim-rec is like P vs EXP for CS. Notably in Ramsey Theory.

Primitive Recursive Functions

Def Prim Rec Functions are in between P and Undecidable.

What They Include Exp, double-exp, Tower, WOWER, etc.

Where Used In some branches of Math Prim-rec vs non-prim-rec is like P vs EXP for CS. Notably in Ramsey Theory.

Verdict Number of Queries as a complexity measure is interesting and could be a theme for the course.

Decidable and Undecidable

Exposition by William Gasarch—U of MD

Decidable Theories

Decidable Theories

1. Presburger Arithmetic is decidable: just $<$ and $+$.

Decidable Theories

1. Presburger Arithmetic is decidable: just $<$ and $+$.
2. WS2S decidable. Uses Tree Automata!

Decidable Theories

1. Presburger Arithmetic is decidable: just $<$ and $+$.
2. WS2S decidable. Uses Tree Automata!
3. S2S is way to hard!

Decidable Theories

1. Presburger Arithmetic is decidable: just $<$ and $+$.
2. WS2S decidable. Uses Tree Automata!
3. S2S is way to hard!
4. Theory of the reals is decidable!

Decidable Theories

1. Presburger Arithmetic is decidable: just $<$ and $+$.
2. WS2S decidable. Uses Tree Automata!
3. S2S is way to hard!
4. Theory of the reals is decidable!

Verdict A step Backwards.

Bounded Queries in Recursion Theory

$\exists$ fnctns computable with 5 queries to HALT but not 4?

Bounded Queries in Recursion Theory

$\exists$ fnctns computable with 5 queries to HALT but not 4?

$\exists$ sets computable with 5 queries to HALT but not with 4?

Bounded Queries in Recursion Theory

$\exists$ fnctns computable with 5 queries to HALT but not 4?

$\exists$ sets computable with 5 queries to HALT but not with 4?

Yes. No assumption needed.

Bounded Queries in Recursion Theory

$\exists$ fnctns computable with 5 queries to HALT but not 4?

$\exists$ sets computable with 5 queries to HALT but not with 4?

Yes. No assumption needed.

Verdict Draws on my own research, so I care. Do you?

More Natural Undecidable Sets

More Natural Undecidable Sets

1. Given a CFG G, is $\overline{L(G)}$ a CFL? Undecidable. Could actually prove this.

More Natural Undecidable Sets

1. Given a CFG G, is $\overline{L(G)}$ a CFL? Undecidable. Could actually prove this.
2. PowerPoint is Undecidable: There is a reduction from HALT to POWERPOINT meaning that if $x \in HALT$ then there will be one slides with a YES, and if $x \notin HALT$ then there will be one slide with a NO. Interesting but too complicated.

More Natural Undecidable Sets

1. Given a CFG G, is $\overline{L(G)}$ a CFL? Undecidable. Could actually prove this.
2. PowerPoint is Undecidable: There is a reduction from HALT to POWERPOINT meaning that if $x \in HALT$ then there will be one slides with a YES, and if $x \notin HALT$ then there will be one slide with a NO. Interesting but too complicated.
3. LaTeX is undecidable. Similar. Why? Because the arithmetic build in to deal with margins is all you need! Interesting but too complicated.

More Natural Undecidable Sets

1. Given a CFG G, is $\overline{L(G)}$ a CFL? Undecidable. Could actually prove this.
2. PowerPoint is Undecidable: There is a reduction from HALT to POWERPOINT meaning that if $x \in HALT$ then there will be one slides with a YES, and if $x \notin HALT$ then there will be one slide with a NO. Interesting but too complicated.
3. LaTeX is undecidable. Similar. Why? Because the arithmetic build in to deal with margins is all you need! Interesting but too complicated.
4. Actually prove Hilbert's tenth is undecidable. Too complicated.

More Natural Undecidable Sets

1. Given a CFG G, is $\overline{L(G)}$ a CFL? Undecidable. Could actually prove this.
2. PowerPoint is Undecidable: There is a reduction from HALT to POWERPOINT meaning that if $x \in HALT$ then there will be one slides with a YES, and if $x \notin HALT$ then there will be one slide with a NO. Interesting but too complicated.
3. LaTeX is undecidable. Similar. Why? Because the arithmetic build in to deal with margins is all you need! Interesting but too complicated.
4. Actually prove Hilbert's tenth is undecidable. Too complicated.

Verdict The first one is plausible, but a step backwards.

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog
WOW There are statements that are true but not provable!

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog
WOW There are statements that are true but not provable!
is like saying

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog
WOW There are statements that are true but not provable!
is like saying
WOW Women can vote!

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog
WOW There are statements that are true but not provable!
is like saying
WOW Women can vote!
Both are true but neither is surprising anymore.

Godel's Incompleteness Theorem

Let T be a theory (e.g., Presburger plus $\times$).
There are theorems that are TRUE but NOT PROVABLE in T.

1. Much easier for us to prove than it was for Godel since we have Turing Machines and know they can do anything that is computable.
2. Important at the time but has been absorbed by the math culture.

Analog
WOW There are statements that are true but not provable!
is like saying
WOW Women can vote!
Both are true but neither is surprising anymore.
Verdict Really not sure about this one. Would need to give context and history, but a very important theorem.

Godel's Second Incompleteness Theorem

Godel's Second Incompleteness Theorem

Tori and Guido in 2036

Godel's Second Incompleteness Theorem

Tori and Guido in 2036

Lets make the statements

Godel's Second Incompleteness Theorem

Tori and Guido in 2036

Lets make the statements

WOW Women can be president!

Godel's Second Incompleteness Theorem

Tori and Guido in 2036

Lets make the statements

WOW Women can be president!

WOW Non-citizen's can be vice-president!

Godel's Second Incompleteness Theorem

Tori and Guido in 2036

Lets make the statements

WOW Women can be president!

WOW Non-citizen's can be vice-president!

True and not surprising.

Arithmetic Hierarchy

Actually **prove** that (say)

$$\mathrm{INF} = \{e : M_e \text{ halts on an infinite number of numbers}\}$$

is NOT in Σ_2.

Arithmetic Hierarchy

Actually **prove** that (say)

$$\mathrm{INF} = \{e : M_e \text{ halts on an infinite number of numbers}\}$$

is NOT in Σ_2.
Verdict Too much background and a step backwards.

Intermediary Sets

Are there sets that are both

Intermediary Sets

Are there sets that are both

1. Not decidable

Intermediary Sets

Are there sets that are both

1. Not decidable
2. Weaker than HALT.

Intermediary Sets

Are there sets that are both

1. Not decidable
2. Weaker than HALT.

Answer: YES and the proof is interesting but hard.

Intermediary Sets

Are there sets that are both

1. Not decidable
2. Weaker than HALT.

Answer: YES and the proof is interesting but hard.
Verdict A step backwards but a very interesting proof.

More Kolmogorov

I could apply Kolm Complexity to

More Kolmogorov

I could apply Kolm Complexity to

1. Proving more langs not regular.

More Kolmogorov

I could apply Kolm Complexity to

1. Proving more langs not regular.
2. Proving some langs have large DFAs, NFAs, CFGs.

More Kolmogorov

I could apply Kolm Complexity to

1. Proving more langs not regular.
2. Proving some langs have large DFAs, NFAs, CFGs.
3. Getting Avg Case Analysis of some algorithms.

Misc

Exposition by William Gasarch—U of MD

Muffins

Muffins

1. Muffin problems have upper and lower bounds that **match**. A good example of what we WANT to be able to achieve in complexity.

Muffins

1. Muffin problems have upper and lower bounds that **match**. A good example of what we WANT to be able to achieve in complexity.
2. My Muffin-Math song:

   `https://www.youtube.com/watch?v=4xQFlsK7jKg`

   is the 2nd worse math song in Youtube. The worst is

Muffins

1. Muffin problems have upper and lower bounds that **match**. A good example of what we WANT to be able to achieve in complexity.
2. My Muffin-Math song:

   `https://www.youtube.com/watch?v=4xQFlsK7jKg`

   is the 2nd worse math song in Youtube. The worst is

   The Bolzano-Weierstrass Rap

   `https://www.youtube.com/watch?v=dfO18klwKHg&t=50s`

Muffins

1. Muffin problems have upper and lower bounds that **match**. A good example of what we WANT to be able to achieve in complexity.
2. My Muffin-Math song:

   `https://www.youtube.com/watch?v=4xQFlsK7jKg`

   is the 2nd worse math song in Youtube. The worst is

   The Bolzano-Weierstrass Rap
   `https://www.youtube.com/watch?v=df018klwKHg&t=50s`
3. The best math song on youtube is about William Rowan Hamilton:
   `https://www.youtube.com/watch?v=SZXHoWwBcDc`

Muffins

1. Muffin problems have upper and lower bounds that **match**. A good example of what we WANT to be able to achieve in complexity.
2. My Muffin-Math song:

   `https://www.youtube.com/watch?v=4xQFlsK7jKg`

   is the 2nd worse math song in Youtube. The worst is

   The Bolzano-Weierstrass Rap

   `https://www.youtube.com/watch?v=dfO18klwKHg&t=50s`
3. The best math song on youtube is about William Rowan Hamilton:

   `https://www.youtube.com/watch?v=SZXHoWwBcDc`

Verdict I want to teach Muffin-Math, Muffin-Math, Muffin-Math, I want to teach Muffin-Math, the answer is 5/12.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$. They want to know if $x = y$.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x
That would take n bits of communication.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x
That would take n bits of communication.
Can they do better?

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x
That would take n bits of communication.
Can they do better? Vote.
1) YES they can and this is known.
2) NO they can't and this is known.
3) UNKNOWN TO SCIENCE.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is x**
That would take n bits of communication.
Can they do better? Vote.
1) YES they can and this is known.
2) NO they can't and this is known.
3) UNKNOWN TO SCIENCE.
NO they can't and this is known.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x
That would take n bits of communication.
Can they do better? Vote.
1) YES they can and this is known.
2) NO they can't and this is known.
3) UNKNOWN TO SCIENCE.
NO they can't and this is known.
Can use results in Comm Complexity to show langs are not regular.

Communication Complexity

Scenario Alice has $x \in \{0,1\}^n$. Bob has $y \in \{0,1\}^n$.
They want to know if $x = y$.
Alice could just say **Hey Bob, my string is** x
That would take n bits of communication.
Can they do better? Vote.
1) YES they can and this is known.
2) NO they can't and this is known.
3) UNKNOWN TO SCIENCE.
NO they can't and this is known.
Can use results in Comm Complexity to show langs are not regular.
Verdict Have done, could do again. A step forward.

Parallel, Randomized, Quantum

There are other modes of computation.

Parallel, Randomized, Quantum

There are other modes of computation.

1. Parallelism: There is a theory analogous to P vs NP to show problems can't be parallelized.

Parallel, Randomized, Quantum

There are other modes of computation.

1. Parallelism: There is a theory analogous to P vs NP to show problems can't be parallelized.
2. Randomized Computations: How much does randomization help?

Parallel, Randomized, Quantum

There are other modes of computation.

1. Parallelism: There is a theory analogous to P vs NP to show problems can't be parallelized.
2. Randomized Computations: How much does randomization help?
3. Quantum Computing: there is a notion of quantum-DFA that I could look into and do, but might be too hard. For me!

Parallel, Randomized, Quantum

There are other modes of computation.

1. Parallelism: There is a theory analogous to P vs NP to show problems can't be parallelized.
2. Randomized Computations: How much does randomization help?
3. Quantum Computing: there is a notion of quantum-DFA that I could look into and do, but might be too hard. For me!

Verdict I would have to look into all of these more to see if they make sense. Quantum would be a step forward.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.
Then we would think CLIQ, TSP, etc were hard.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.
Then we would think CLIQ, TSP, etc were hard.

There are other groups of problems where this IS what we have.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.
Then we would think CLIQ, TSP, etc were hard.

There are other groups of problems where this IS what we have.

1. All-pairs-shortest-path seems to REQUIRE $\Omega(n^2)$ time. There are now APSP-hard problems.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.
Then we would think CLIQ, TSP, etc were hard.

There are other groups of problems where this IS what we have.

1. All-pairs-shortest-path seems to REQUIRE $\Omega(n^2)$ time. There are now APSP-hard problems.
2. CLIQ_k seems to REQUIRE $n^{\Omega(k)}$ time. There are now CLIQ_k-hard problems.

Complexity Classes Based on Problems

Imagine if we did not have Cook-Levin but still thought SAT was hard.
Then we would think CLIQ, TSP, etc were hard.

There are other groups of problems where this IS what we have.

1. All-pairs-shortest-path seems to REQUIRE $\Omega(n^2)$ time. There are now APSP-hard problems.
2. CLIQ_k seems to REQUIRE $n^{\Omega(k)}$ time. There are now CLIQ_k-hard problems.
3. There are others.

What to take Out (Brief)

Exposition by William Gasarch—U of MD

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

1. CFG's I could easily take out. :-)

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

1. CFG's I could easily take out. :-)
2. Recursion Theory. NEED to prove HALT is undecidable. LIKE to prove WS1S is decidable. All else can go. Maybe even WS1S can go :-(

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

1. CFG's I could easily take out. :-)
2. Recursion Theory. NEED to prove HALT is undecidable. LIKE to prove WS1S is decidable. All else can go. Maybe even WS1S can go :-(
3. Could reduce how much time I spend on regular by cutting out Regular Expressions. They are done in 330 anyway. DO want to keep the SMALL-NFA-RESPECT problem.

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

1. CFG's I could easily take out. :-)
2. Recursion Theory. NEED to prove HALT is undecidable. LIKE to prove WS1S is decidable. All else can go. Maybe even WS1S can go :-(
3. Could reduce how much time I spend on regular by cutting out Regular Expressions. They are done in 330 anyway. DO want to keep the SMALL-NFA-RESPECT problem.
4. Could do less HW review- only go over the problems student had trouble with.

What to Take Out?

If I want to put any of that in, I need to take some stuff out.

1. CFG's I could easily take out. :-)
2. Recursion Theory. NEED to prove HALT is undecidable. LIKE to prove WS1S is decidable. All else can go. Maybe even WS1S can go :-(
3. Could reduce how much time I spend on regular by cutting out Regular Expressions. They are done in 330 anyway. DO want to keep the SMALL-NFA-RESPECT problem.
4. Could do less HW review- only go over the problems student had trouble with.
5. Could go faster by making it a truly flipped classroom.

BILL AND NATHAN RECORD LECTURE!!!!

BILL AND NATHAN RECORD LECTURE!!!

UN-TIMED PART OF FINAL IS TUESDAY May 11 11:00A. NO DEAD CAT

Exposition by William Gasarch—U of MD

FINAL IS THURSDAY May 13 8:00PM-10:15PM

Exposition by William Gasarch—U of MD

FILL OUT COURSE EVALS for ALL YOUR COURSES!!!

Exposition by William Gasarch—U of MD