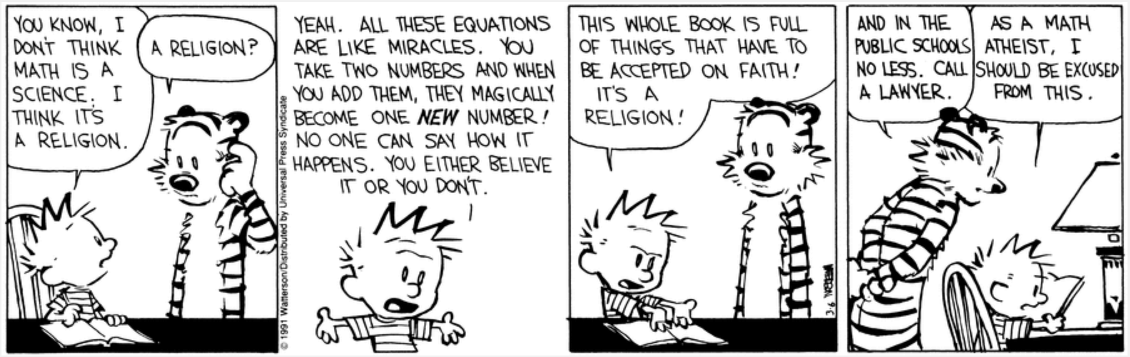
Welcome to CS420!

Intro to Theory of Computation

UMass Boston Computer Science

Instructor: Stephen Chang

Spring 2022


YOU KNOW, I DON'T THINK MATH IS A SCIENCE. I THINK IT'S A RELIGION.
A RELIGION?
YEAH. ALL THESE EQUATIONS ARE LIKE MIRACLES. YOU TAKE TWO NUMBERS AND WHEN YOU ADD THEM, THEY MAGICALLY BECOME ONE NEW NUMBER! NO ONE CAN SAY HOW IT HAPPENS. YOU EITHER BELIEVE IT OR YOU DON'T.
THIS WHOLE BOOK IS FULL OF THINGS THAT HAVE TO BE ACCEPTED ON FAITH! IT'S A RELIGION!
AND IN THE PUBLIC SCHOOLS NO LESS. CALL A LAWYER.
AS A MATH ATHEIST, I SHOULD BE EXCUSED FROM THIS.
© 1991 Watterson/Distributed by Universal Press Syndicate

Welcome to CS420!

Intro to Theory of **Computation**

What's this?

UMass Boston Computer Science

Instructor: Stephen Chang

Spring 2022

CS 420 Lecture Logistics

- I expect lecture to be interactive
  - It’s the best way to learn
  - (Participation is a part of grade)

- I may call on students randomly
  - It’s ok to be wrong

- Please state your name before speaking
  - It will help me get to know all of you

Welcome to CS420!

Intro to Theory of Computation

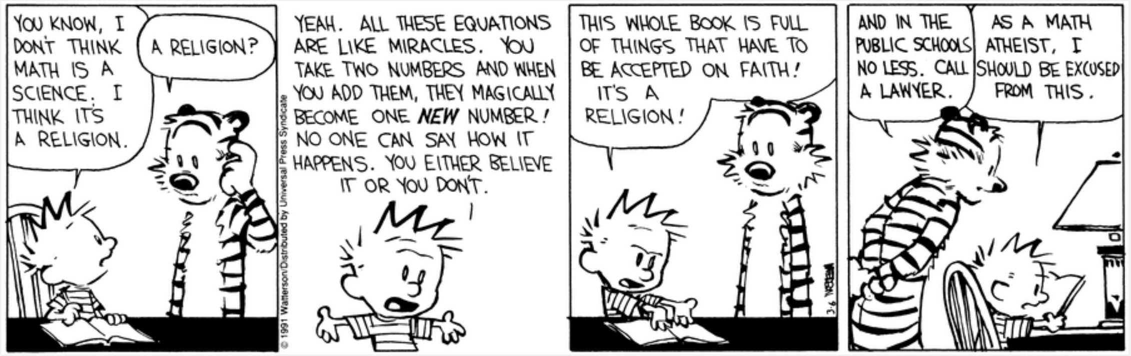
How would you define this?

UMass Boston Computer Science

Instructor: Stephen Chang

Spring 2022

Computation Is …


YOU KNOW, I DON'T THINK MATH IS A SCIENCE. I THINK IT'S A RELIGION.
A RELIGION?
YEAH. ALL THESE EQUATIONS ARE LIKE MIRACLES. YOU TAKE TWO NUMBERS AND WHEN YOU ADD THEM, THEY MAGICALLY BECOME ONE *NEW* NUMBER! NO ONE CAN SAY HOW IT HAPPENS. YOU EITHER BELIEVE IT OR YOU DON'T.
THIS WHOLE BOOK IS FULL OF THINGS THAT HAVE TO BE ACCEPTED ON FAITH! IT'S A RELIGION!
AND IN THE PUBLIC SCHOOLS NO LESS. CALL A LAWYER.
AS A MATH ATHEIST, I SHOULD BE EXCUSED FROM THIS.
© 1991 Watterson/Distributed by Universal Press Syndicate

- 1 + 1 = ??
- = 2

… some base definitions and assumptions ("axioms") …

- 11 + 11 = ??
- = 22

… and rules for using those initial definitions and axioms ("algorithm")

- 55 + 55 = ??
- = 110

Rules can be executed by hand, or by a machine

- 1 +1 = ??
- = 10

There are many possible definitions of **computation**.

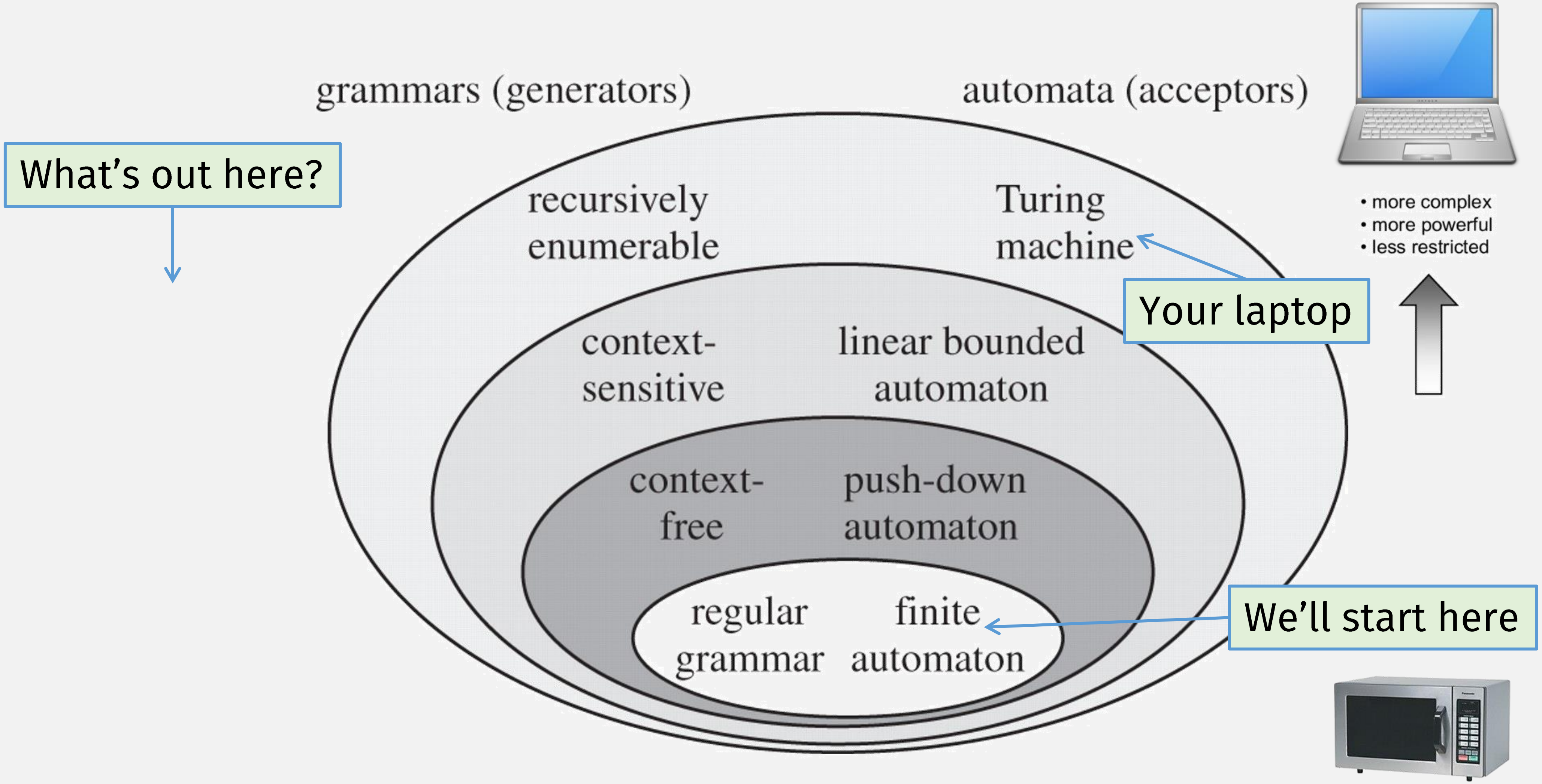
Many Different Kinds of Computation

How do they relate to each other?

This class is about:

- formally defining computer(s) and computation,
- and studying their relation to each other!

Many Levels of Computation



grammars (generators)

automata (acceptors)

What's out here?

recursively enumerable

Turing machine

Your laptop

context-sensitive

linear bounded automaton

context-free

push-down automaton

regular grammar

finite automaton

We'll start here

- more complex
- more powerful
- less restricted

Welcome to CS420!

Intro to Theory of Computation & Computers

UMass Boston Computer Science

Instructor: Stephen Chang

Spring 2022

Welcome to CS420!

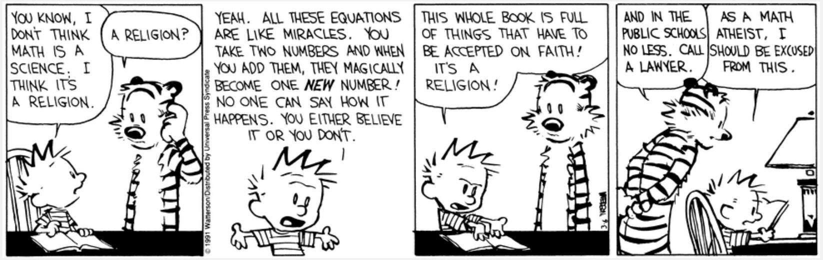
Intro to **Theory** of Computation & Computers

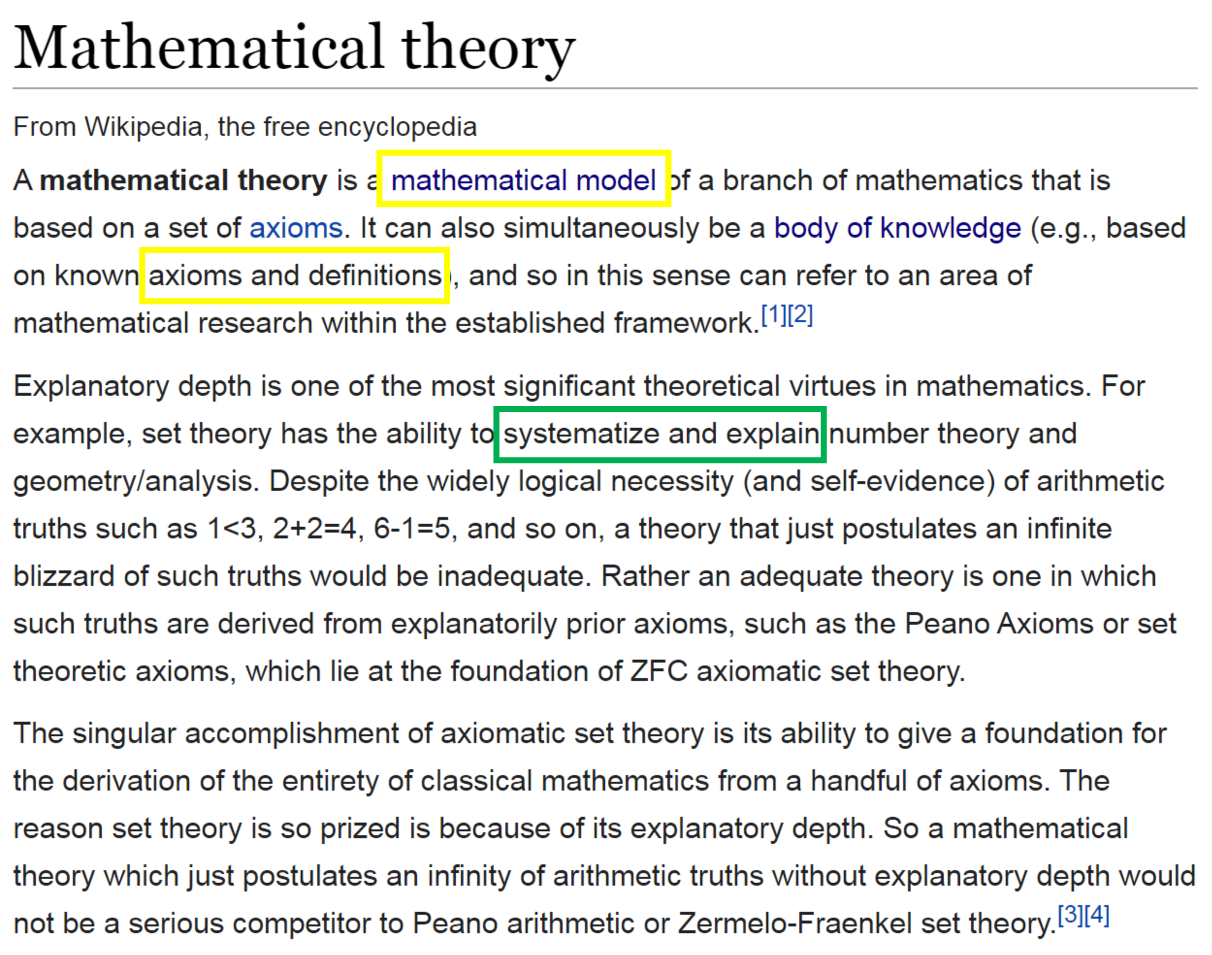
This class is about **formally** defining computer(s) and computation!

“formally” = mathematically

(This is a math course!)

A (Mathematical) Theory Is …


YOU KNOW, I DON'T THINK MATH IS A SCIENCE. I THINK IT'S A RELIGION.
A RELIGION?
YEAH. ALL THESE EQUATIONS ARE LIKE MIRACLES. YOU TAKE TWO NUMBERS AND WHEN YOU ADD THEM, THEY MAGICALLY BECOME ONE *NEW* NUMBER! NO ONE CAN SAY HOW IT HAPPENS. YOU EITHER BELIEVE IT OR YOU DON'T.
THIS WHOLE BOOK IS FULL OF THINGS THAT HAVE TO BE ACCEPTED ON FAITH! IT'S A RELIGION!
AND IN THE PUBLIC SCHOOLS NO LESS. CALL A LAWYER.
AS A MATH ATHEIST, I SHOULD BE EXCUSED FROM THIS.
© 1991 Watterson/Distributed by Universal Press Syndicate
3-6 WATTERSON



Mathematical theory

From Wikipedia, the free encyclopedia

A **mathematical theory** is a mathematical model of a branch of mathematics that is based on a set of axioms. It can also simultaneously be a body of knowledge (e.g., based on known axioms and definitions, and so in this sense can refer to an area of mathematical research within the established framework.[1][2]

Explanatory depth is one of the most significant theoretical virtues in mathematics. For example, set theory has the ability to systematize and explain number theory and geometry/analysis. Despite the widely logical necessity (and self-evidence) of arithmetic truths such as 1<3, 2+2=4, 6-1=5, and so on, a theory that just postulates an infinite blizzard of such truths would be inadequate. Rather an adequate theory is one in which such truths are derived from explanatorily prior axioms, such as the Peano Axioms or set theoretic axioms, which lie at the foundation of ZFC axiomatic set theory.

The singular accomplishment of axiomatic set theory is its ability to give a foundation for the derivation of the entirety of classical mathematics from a handful of axioms. The reason set theory is so prized is because of its explanatory depth. So a mathematical theory which just postulates an infinity of arithmetic truths without explanatory depth would not be a serious competitor to Peano arithmetic or Zermelo-Fraenkel set theory.[3][4]

… a mathematical model,
i.e., **axioms** and **definitions**, of
some domain, e.g. computers …

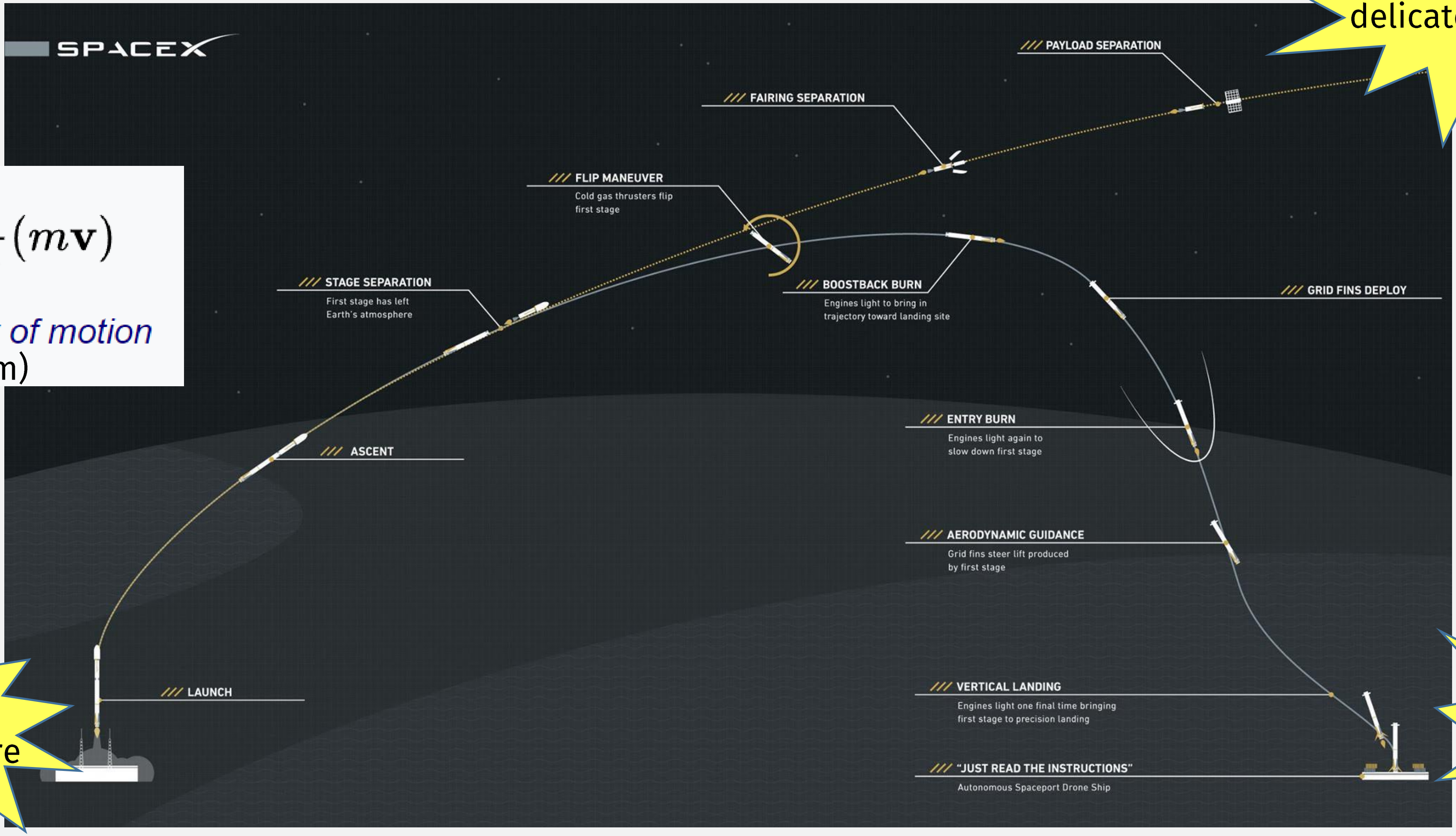
… that **explains (predicts)**
some real-world phenomena …

Example: Theory of Classical Mechanics

$$\mathbf{F} = \frac{d}{dt}(m\mathbf{v})$$

Second law of motion
(axiom)

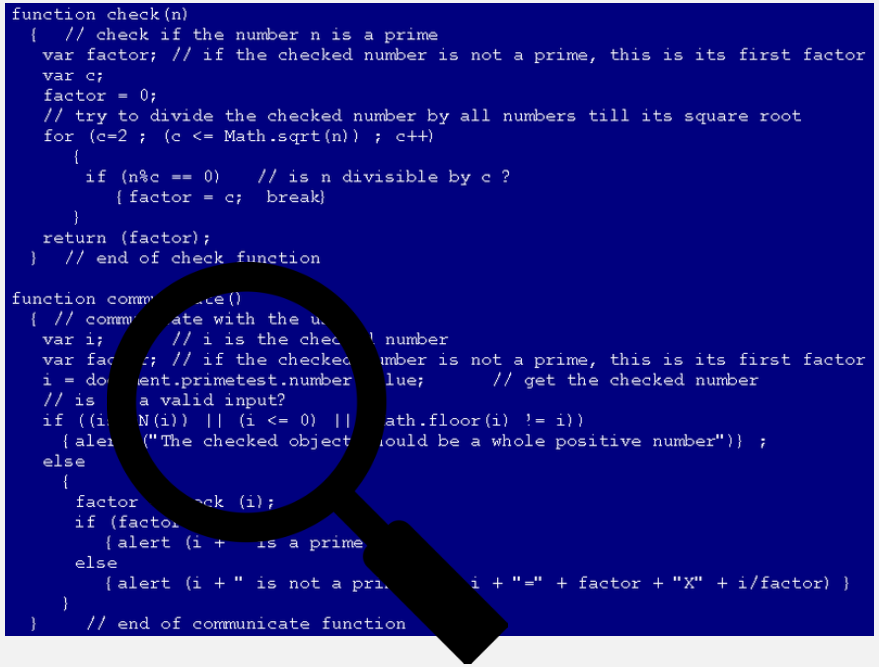
Drop off $1b of delicate stuff here



Do 1000 ton explosion here

Land exactly here

Why make predictions about computers?

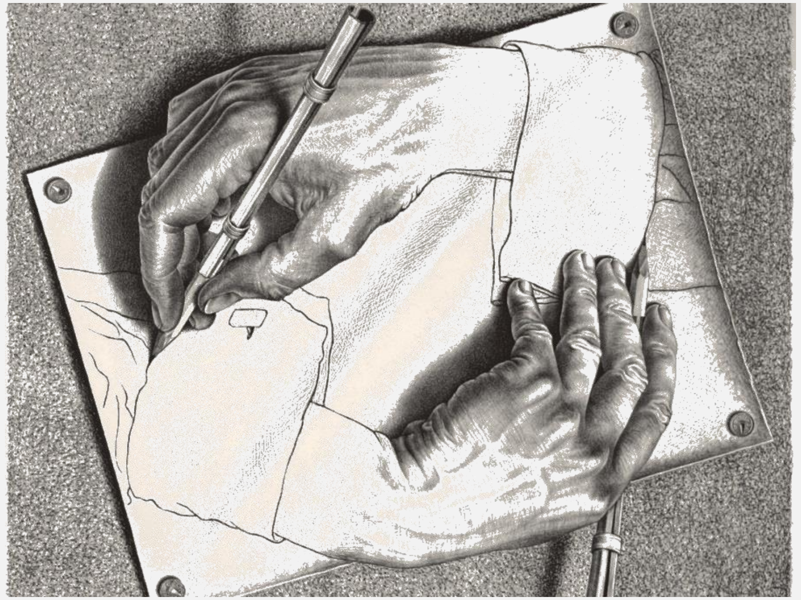


```
function check(n)
 {  // check if the number n is a prime
   var factor; // if the checked number is not a prime, this is its first factor
   var c;
   factor = 0;
   // try to divide the checked number by all numbers till its square root
   for (c=2 ; (c <= Math.sqrt(n)) ; c++)
      {
       if (n%c == 0)    // is n divisible by c ?
          { factor = c;  break}
      }
   return (factor);
 }  // end of check function

function comm    ce()
 { // comm   ate with the u
   var i;      // i is the chec   number
   var fac  ; // if the checked  umber is not a prime, this is its first factor
   i = doc  ent.primetest.number  lue;       // get the checked number
   // is   a valid input?
   if ((i  N(i)) || (i <= 0) ||   ath.floor(i) != i))
     { aler  ("The checked object  ould be a whole positive number")} ;
   else
     {
      factor    ck (i);
      if (facto
         { alert (i +   is a prime
      else
         { alert (i + " is not a pri      i + "=" + factor + "X" + i/factor) }
     }
 }    // end of communicate function
```

Predict result without running a program?

Can we make predictions about computers?



Trying to predict computation requires computation!

Can we make predictions about computers?

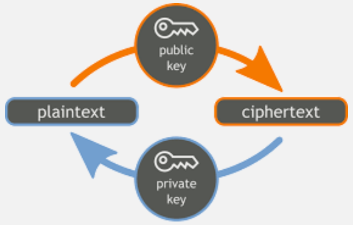
- The **Halting Lemma** says:
- And **Rice’s Theorem** says:

  - “all non-trivial, semantic properties of programs are undecidable”
- Actually:
  - it depends on the computation model!

Knowing What Computers Can't Do is Still Useful!

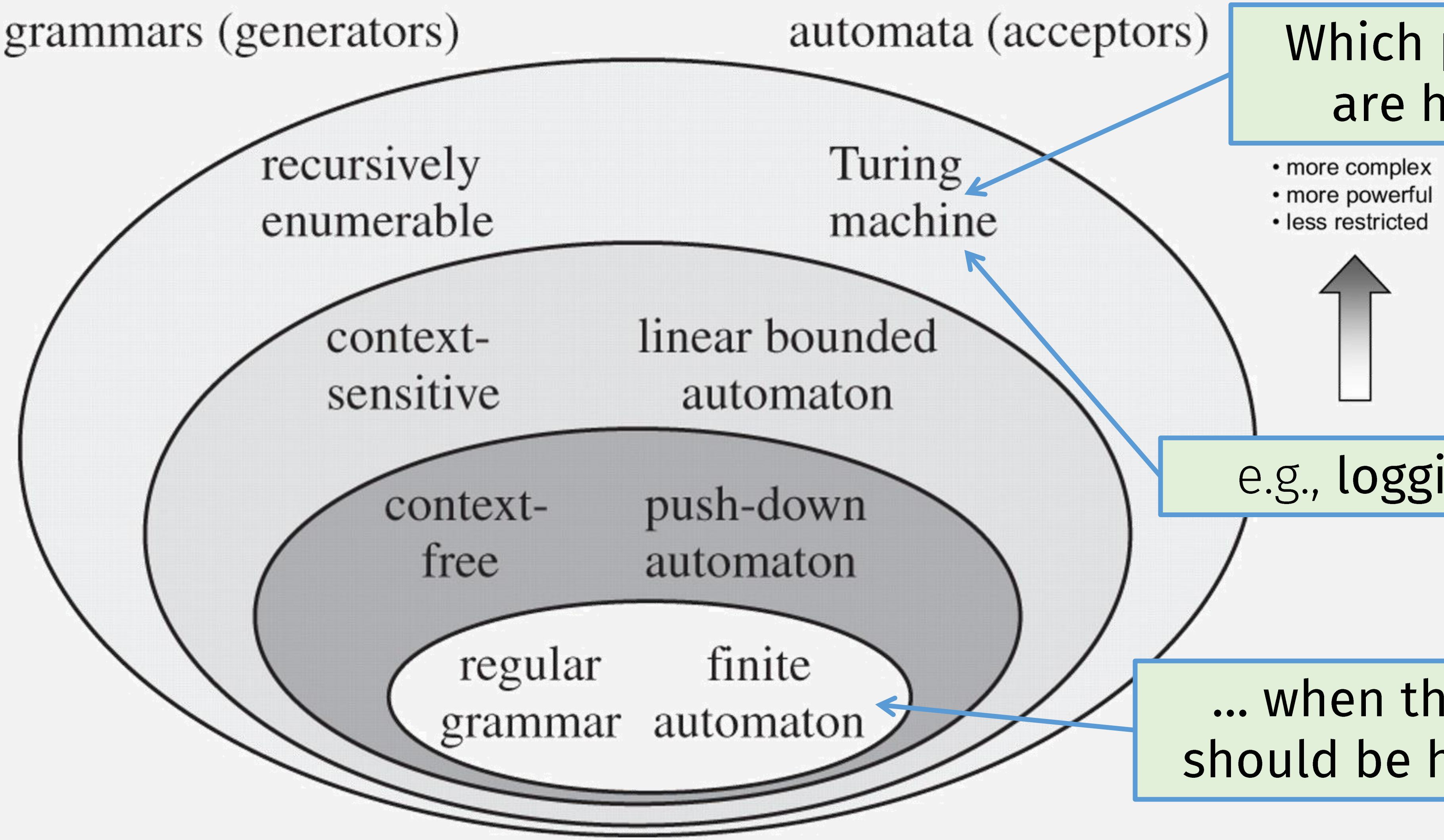
In Cryptography:

- Perfect secrecy is impossible in practice
- But with slightly imperfect secrecy (i.e., a computationally bounded adversary):



- But there are still problems, even with strong mathematical foundations:
  - with users, implementors **who don't understand theory of computation**

Programs running programs: How much power to give?

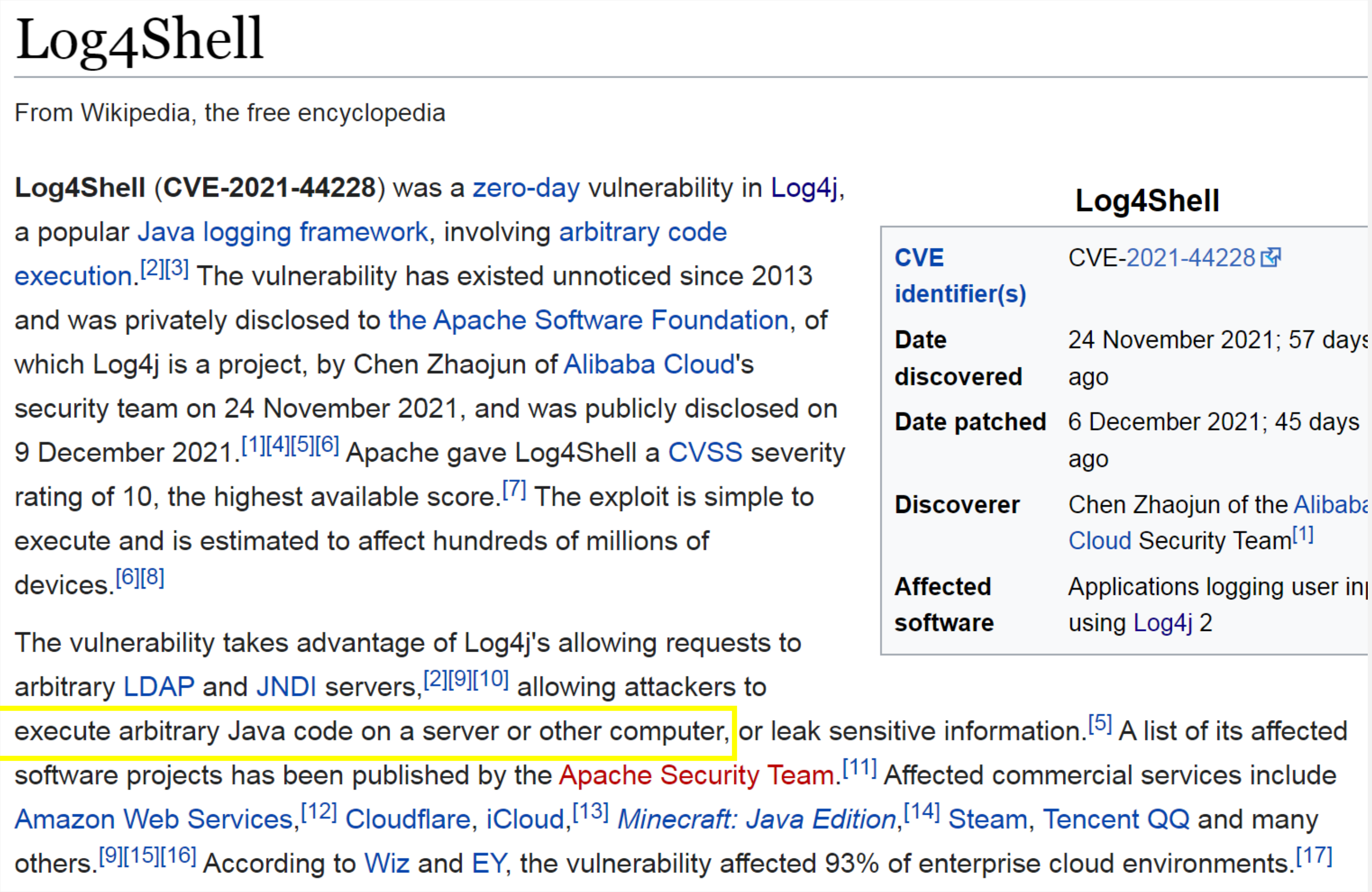


Which programs are here … ?

e.g., **logging?**

… when they should be here

The Computing Power of Logs?



Log4Shell

From Wikipedia, the free encyclopedia

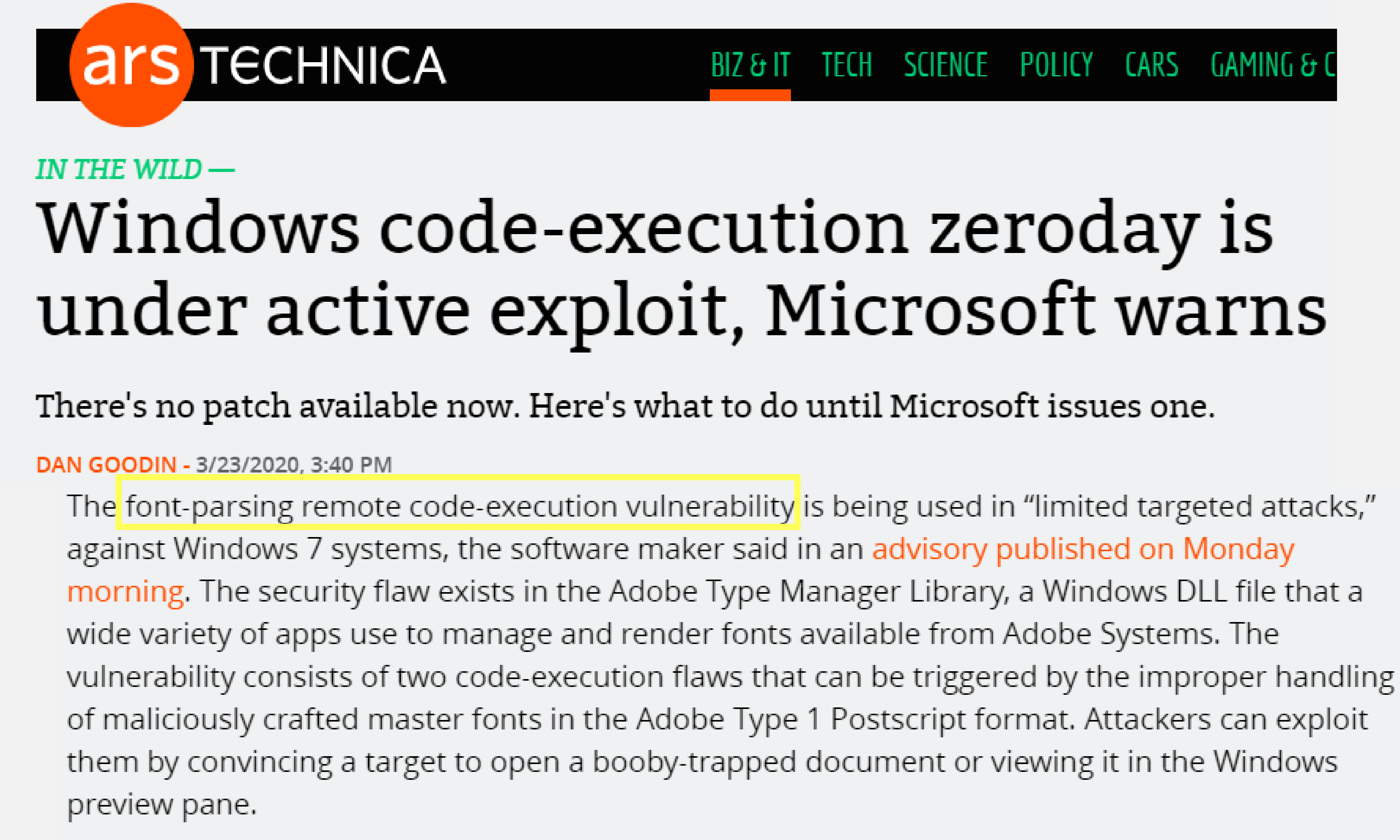
Log4Shell (**CVE-2021-44228**) was a zero-day vulnerability in Log4j, a popular Java logging framework, involving arbitrary code execution.[2][3] The vulnerability has existed unnoticed since 2013 and was privately disclosed to the Apache Software Foundation, of which Log4j is a project, by Chen Zhaojun of Alibaba Cloud's security team on 24 November 2021, and was publicly disclosed on 9 December 2021.[1][4][5][6] Apache gave Log4Shell a CVSS severity rating of 10, the highest available score.[7] The exploit is simple to execute and is estimated to affect hundreds of millions of devices.[6][8]

Log4Shell

CVE identifier(s)	CVE-2021-44228
Date discovered	24 November 2021; 57 days ago
Date patched	6 December 2021; 45 days ago
Discoverer	Chen Zhaojun of the Alibaba Cloud Security Team[1]
Affected software	Applications logging user in using Log4j 2

The vulnerability takes advantage of Log4j's allowing requests to arbitrary LDAP and JNDI servers,[2][9][10] allowing attackers to execute arbitrary Java code on a server or other computer, or leak sensitive information.[5] A list of its affected software projects has been published by the Apache Security Team.[11] Affected commercial services include Amazon Web Services,[12] Cloudflare, iCloud,[13] *Minecraft: Java Edition*,[14] Steam, Tencent QQ and many others.[9][15][16] According to Wiz and EY, the vulnerability affected 93% of enterprise cloud environments.[17]

The Computing Power of Fonts?



IN THE WILD —

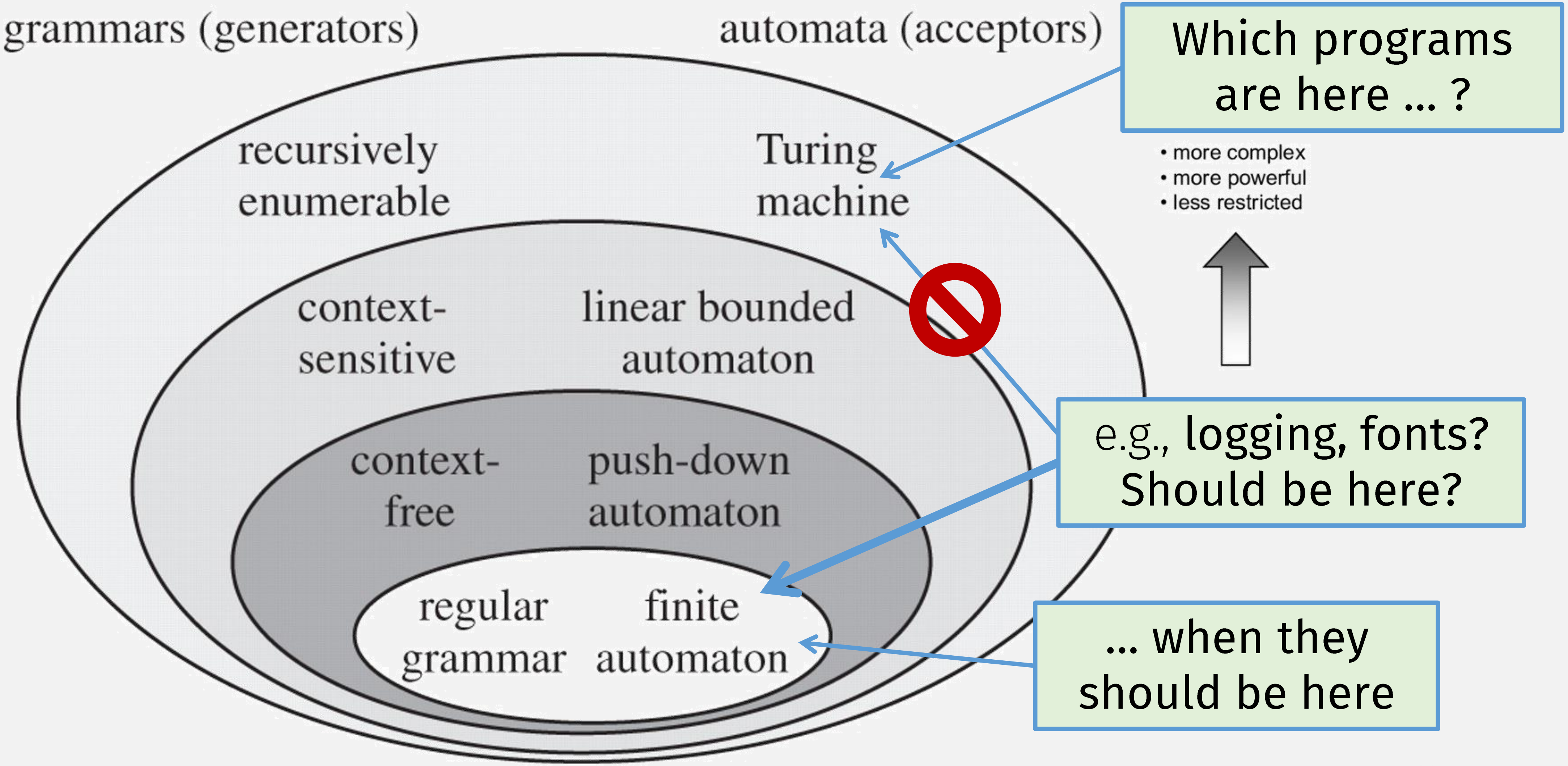
Windows code-execution zeroday is under active exploit, Microsoft warns

There's no patch available now. Here's what to do until Microsoft issues one.

DAN GOODIN - 3/23/2020, 3:40 PM

The font-parsing remote code-execution vulnerability is being used in “limited targeted attacks,” against Windows 7 systems, the software maker said in an advisory published on Monday morning. The security flaw exists in the Adobe Type Manager Library, a Windows DLL file that a wide variety of apps use to manage and render fonts available from Adobe Systems. The vulnerability consists of two code-execution flaws that can be triggered by the improper handling of maliciously crafted master fonts in the Adobe Type 1 Postscript format. Attackers can exploit them by convincing a target to open a booby-trapped document or viewing it in the Windows preview pane.

Programs running programs: How much power to give?

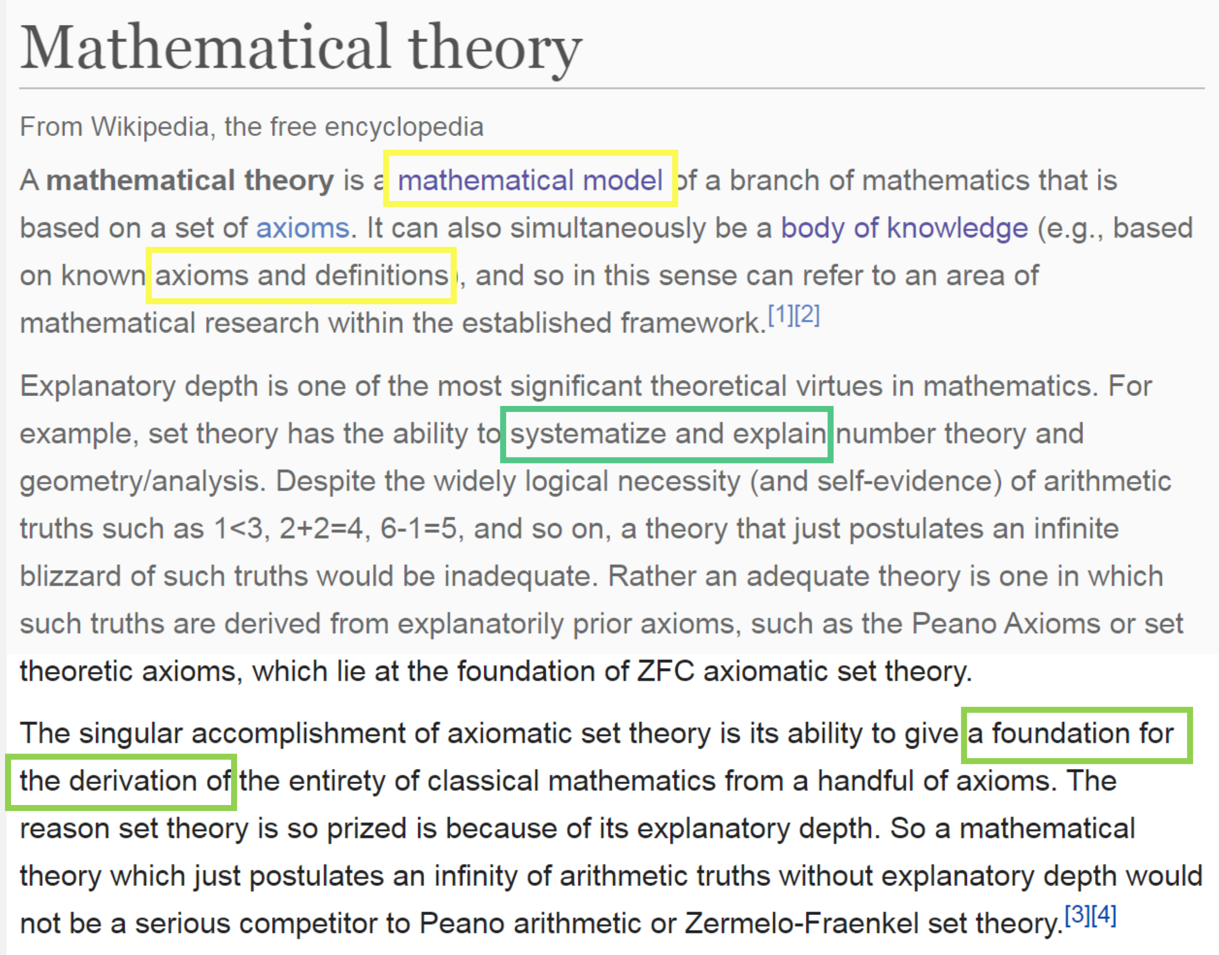


Which programs are here … ?

e.g., logging, fonts? Should be here?

… when they should be here

A (Mathematical) Theory Is …



Mathematical theory

From Wikipedia, the free encyclopedia

A **mathematical theory** is a mathematical model of a branch of mathematics that is based on a set of axioms. It can also simultaneously be a body of knowledge (e.g., based on known axioms and definitions, and so in this sense can refer to an area of mathematical research within the established framework.[1][2]

Explanatory depth is one of the most significant theoretical virtues in mathematics. For example, set theory has the ability to systematize and explain number theory and geometry/analysis. Despite the widely logical necessity (and self-evidence) of arithmetic truths such as 1<3, 2+2=4, 6-1=5, and so on, a theory that just postulates an infinite blizzard of such truths would be inadequate. Rather an adequate theory is one in which such truths are derived from explanatorily prior axioms, such as the Peano Axioms or set theoretic axioms, which lie at the foundation of ZFC axiomatic set theory.

The singular accomplishment of axiomatic set theory is its ability to give a foundation for the derivation of the entirety of classical mathematics from a handful of axioms. The reason set theory is so prized is because of its explanatory depth. So a mathematical theory which just postulates an infinity of arithmetic truths without explanatory depth would not be a serious competitor to Peano arithmetic or Zermelo-Fraenkel set theory.[3][4]

… a mathematical model, i.e., **axioms** and **definitions**, of some domain, e.g. computers …

… that **explains (predicts)** some real-world phenomena …

… and can **derive** additional results (**lemmas** & **theorems**) …

How Mathematics Works

Mathematician
(or student)

Actually, **it's not always so easy to create the next level … Preciseness is important**

Proofs = Figuring out how to (precisely) stack the pieces together

More Theorems

More Axioms

More Definitions

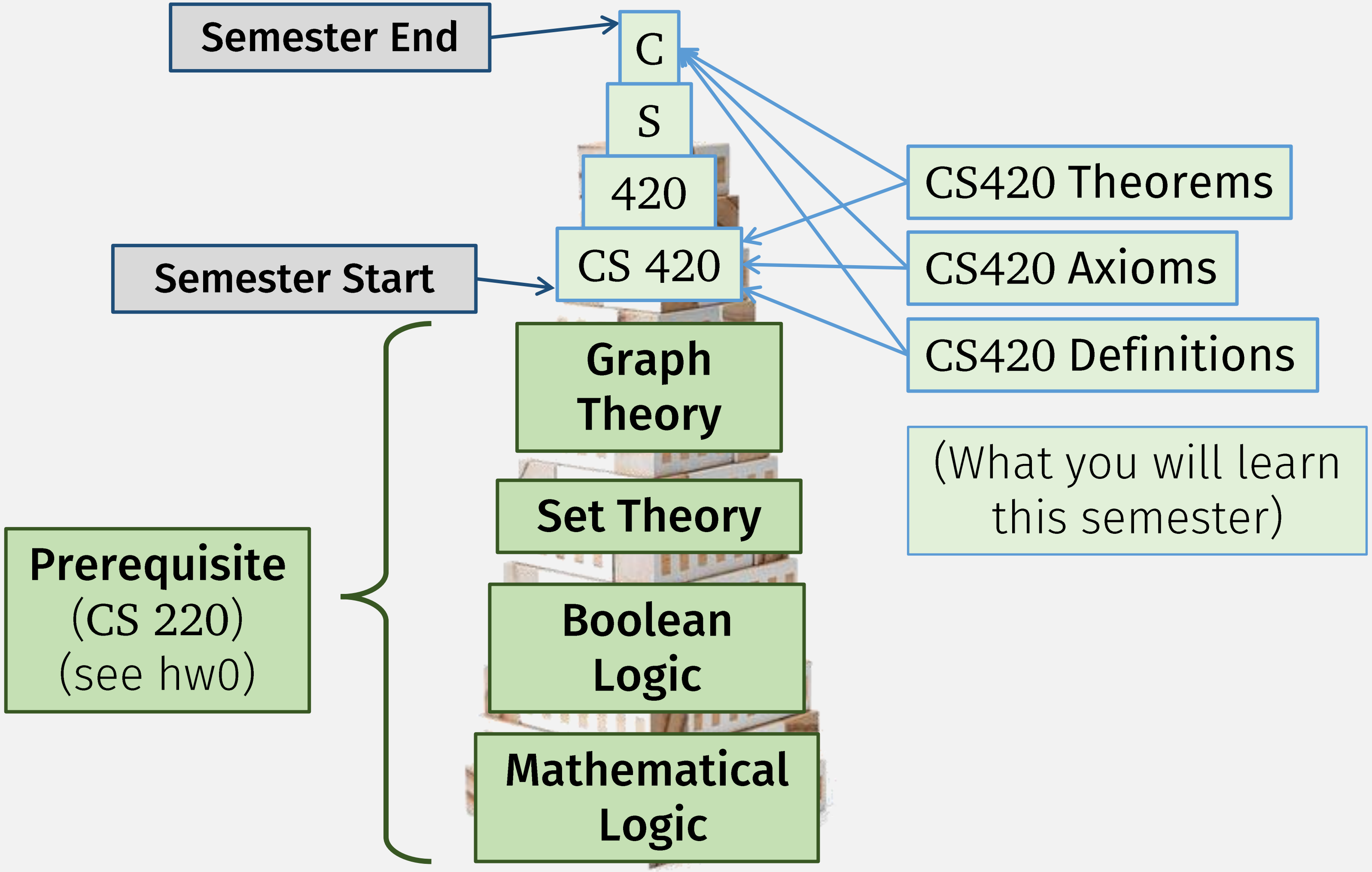
Theorem

Theorem

Axioms

Definitions

How **CS 420** Works



Semester End

C

S

420

CS 420

Semester Start

CS420 Theorems

CS420 Axioms

CS420 Definitions

(What you will learn this semester)

Graph Theory

Set Theory

Boolean Logic

Mathematical Logic

Prerequisite
(CS 220)
(see hw0)

Another Analogy

Remember:

Preciseness is important

(Proofs must connect things together exactly)

Semester Start

Must get these items …

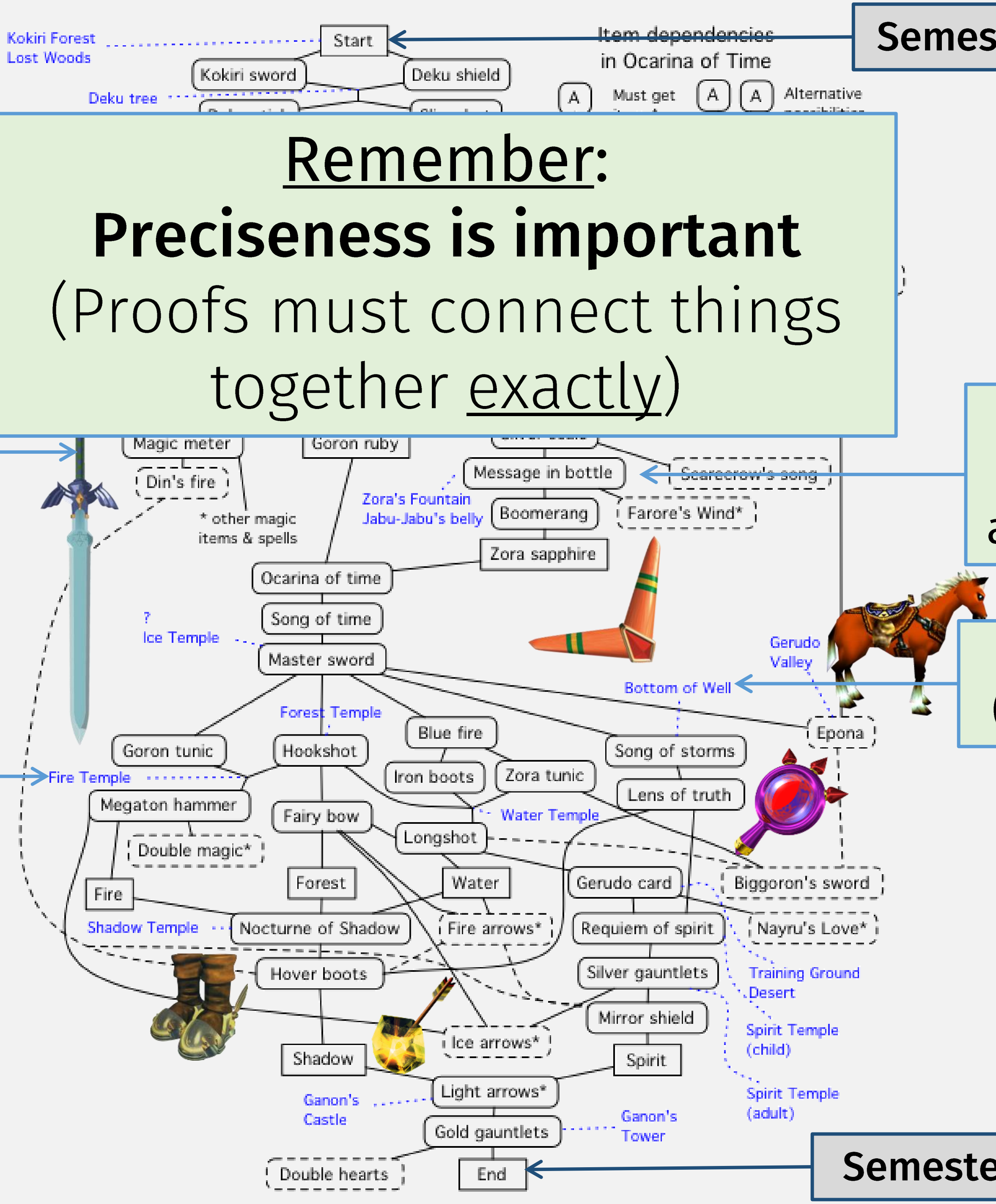
… to finish this quest

“inventory”

I.e., Need to know **axioms, definitions,** and (prev) **theorems …**

… to prove a (new) theorem

Semester End

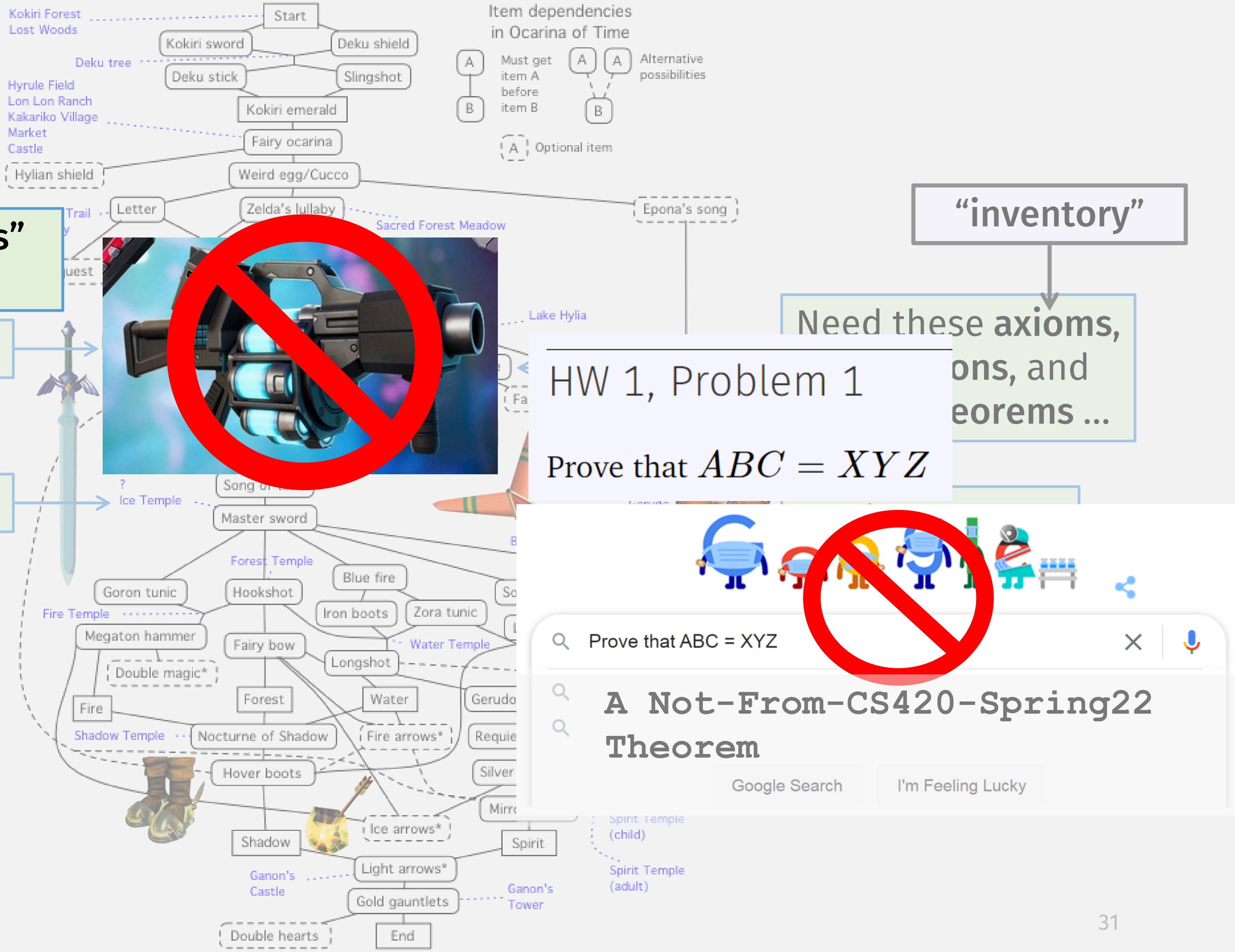


Word of Caution

External “items” won’t help

Must get these items ...

... to finish this quest



“inventory”

Need these **axioms**, ...ons, and ...eorems ...

HW 1, Problem 1

Prove that $ABC = XYZ$

Prove that ABC = XYZ

A Not-From-CS420-Spring22 Theorem

How to Do Well in this Course

- Learn the “inventory”
  - I.e., axioms, definitions, and theorems

- To solve a problem (prove a new theorem), think about how to precisely combine and use things from the “inventory”

- Don’t Fall Behind!
  - Start HW Early (HW 0 due Sunday 11:59pm EST)

- Participate and Engage
  - Lecture
  - Office Hours
  - Message Boards

Textbooks

- Sipser. *Intro to Theory of Computation*, 3rd ed.
- Hopcroft, Motwani, Ullman. *Intro to Automata Theory, Languages, and Computation*, 3rd ed.

- Recommended but not required,
- slides and lecture should be self-contained,
- Readings to accompany lectures will be posted

All course info available on web site:
`https://www.cs.umb.edu/~stchang/cs420/s22`

Grading

- **HW**: 80%
  - Weekly: Out Monday, In Sunday
  - Approx. 12 assignments
  - Lowest grade dropped
- **Quizzes**: 5%
  - End of every lecture
  - To help everyone keep up
- **Participation**: 15%
  - Lecture, office hours, piazza
- No exams

- **A** range: 90-100
- **B** range: 80-90
- **C** range: 70-80
- **D** range: 60-70
- **F**: < 60

All course info available on web site:
`https://www.cs.umb.edu/~stchang/cs420/s22`

Late HW

- Is bad … try not to do it please
  - Grades get delayed
  - Can’t discuss solutions
  - Makes it hard to catch up!

- Late Policy: **3 late days** to use during the semester

HW Collaboration Policy

Allowed

- Discussing HW with classmates
- Using other resources, e.g., youtube, other texts, etc.
- Writing up answers <u>on your own</u>, from scratch, in your own words

Not Allowed

- Submitting someone else's answer
- It's still someone else's answer if:
  - changing variables,
  - cutting words,
  - or rearranging sentences …
- Using sites like Chegg, etc.
- Using "inventory" <u>not from this course</u>

Honesty Policy

- 1st offense: zero on problem
- 2nd offense: zero on hw, reported to school
- 3rd offense+: F for course

Regret policy

- If you self-report an honesty violation, you’ll only receive a zero on the problem and the issue will be immediately resolved (don’t abuse this please).

All Up to Date Course Info

Survey, Schedule, Office Hours, HWs, …

See course website:

https://www.cs.umb.edu/~stchang/cs420/s22/

Check-In Quiz 1/24

(see gradescope)