CSE215: Lecture 1
Foundations of Computer Science

Instructor: Zhoulai Fu

State University of New York, Korea

February 21, 2022

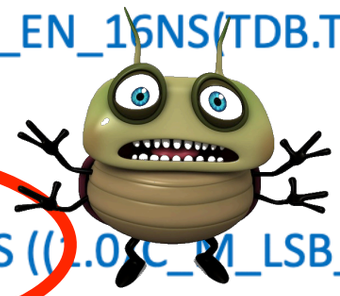
Course materials and Info available here:
https://github.com/zhoulaifu/22_cse215_spring

French Ariane 5 Rocket, 1996

What is Floating-Point ?

Ada code for Ariane 5 Rocket

```
if L_M_BV_32 > 32767 then
  P_M_DERIVE(T_ALG.E_BV) := 16#7FFF#;
elsif L_M_BV_32 < -32768 then
  P_M_DERIVE(T_ALG.E_BV) := 16#8000#;
else
  P_M_DERIVE(T_ALG.E_BV) := UC_16S_EN_16NS(TDB.T_ENTIER_16S(L_M_BV_32));
end if;
P_M_DERIVE(T_ALG.E_BH) :=
  UC_16S_EN_16NS (TDB.T_ENTIER_16S ((1.0/C_M_LSB_BH)*G_M_INFO_DERIVE(T_ALG.E_BH)));
```



$7 billion Software Disaster

Comparison:

SUNY Korea was awarded $0.05 billion for 10 years under an MKE grant
(Source: https://sunyk.cs.stonybrook.edu/)

SUNY, with 64 campus, has $7.43 billion revenues in 2021
(source: 2021 Annual Financial Report - SUNY)

From 2018 to 2020, South Korea GDP dropped $94 billion;
(Source: World bank)

Propositional Logic

Predicate Logic

Proof

Why does a computing system fail (or work)?

Sequences

Sets

Functions

Relations

Expected Learning Outcomes

- An ability to check if a mathematical argument is valid and sound
- An ability to verify the correctness of proofs of some existing theorems and prove some new theorems
- An ability to use the mathematical concepts of sequences, functions, relations, and sets in solving computing problems

Meet the Instructor

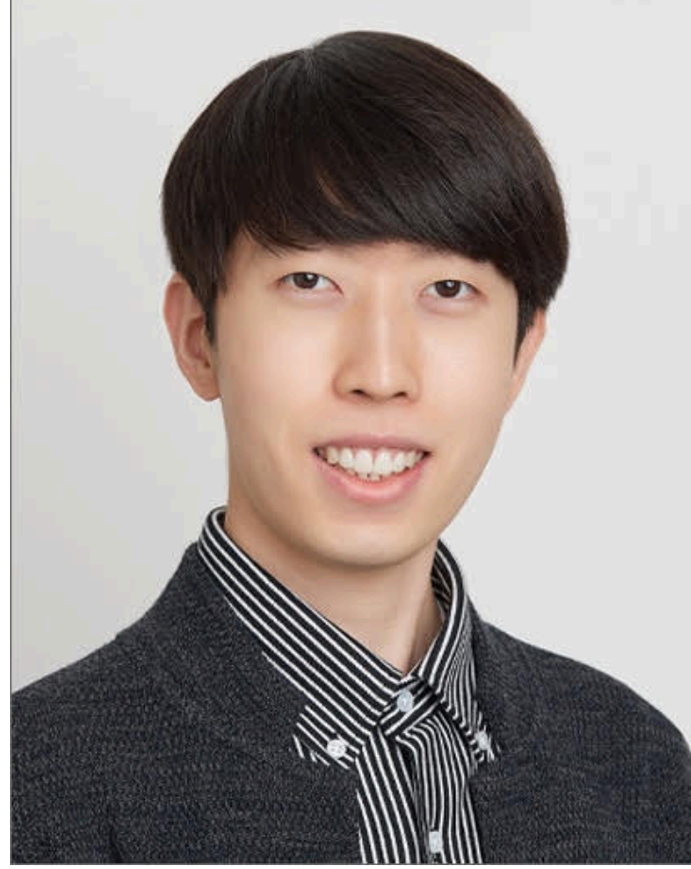
Education

- B.Sc, M.Sc, Ecole Polytechnique, France
- M.Eng. Telecom Paris, France
- Ph.D. INRIA (National CS Lab), France

Teaching & Research

- University of California Davis, United States
- IT University of Copenhagen, Denmark
- SUNY Korea

Meet our TA: Minki Jeon



Education

- SUNY Korea (Transferred from GSU), CS Major
- Georgia State University, CS Major

Teaching & Research

- Member of SUNY Korea Co;Ders Us
- Member of SUNY Korea Computing Society (SKSC)
- TA of CSE 114 (2020 Fall semester)
- Currently working as a coding instructor at a private educational institution
- Currently working as a web frontend developer

Each team member's jobs

You	TA	Instructor
Lectures	**Office hours**	**Office hours**
Homework	**Grading HW**	**Lectures**
		Recitations
		Grading
Forum	**Forum**	**Forum**

But the most important: do our best to stay healthy

Practical matters

- COVID
- Texbook
- Schedule
- Homework
- Exams and grading
- Need help

COVID ptotocol as of today

- If a student develops symptom(s) and/or is confirmed to be infected with COVID-19 before class, she/he is required to report that to the instructors of all of her/his courses immediately, but no later than the start of the classes.

- Those students must not attend all the in-person classes he/she is enrolled in for one week (if confirmed) or be tested (in case of showing symptoms) prior to returning to class.

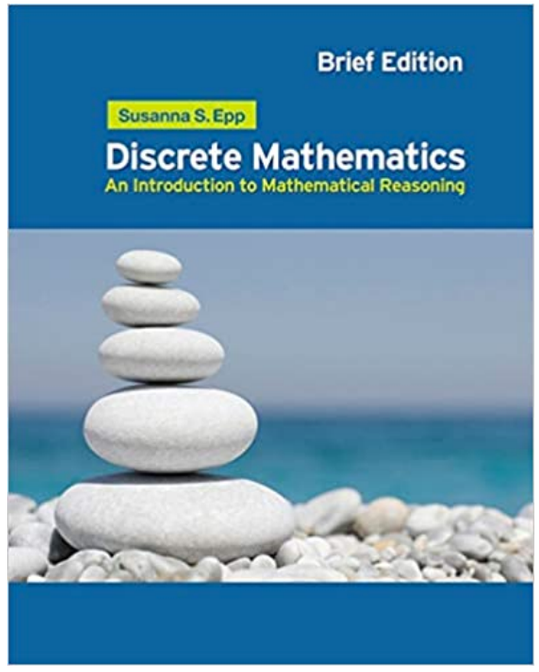
- Instructors must report all of the confirmed case(s), including their own, to SUNY Korea's COVID-19 Task Force at covid19report@sunykorea.ac.kr immediately when a confirmed case is found.

Two-Stage Contingency Plan

- Stage 1. If the rate of the confirmed case surpasses 5% of the members of the university community, most classes except essential major or lab courses* should be switched to online.

- Stage 2. If the rate of the confirmed case surpasses 10% of the university community all classes should be provided online.

Textbook



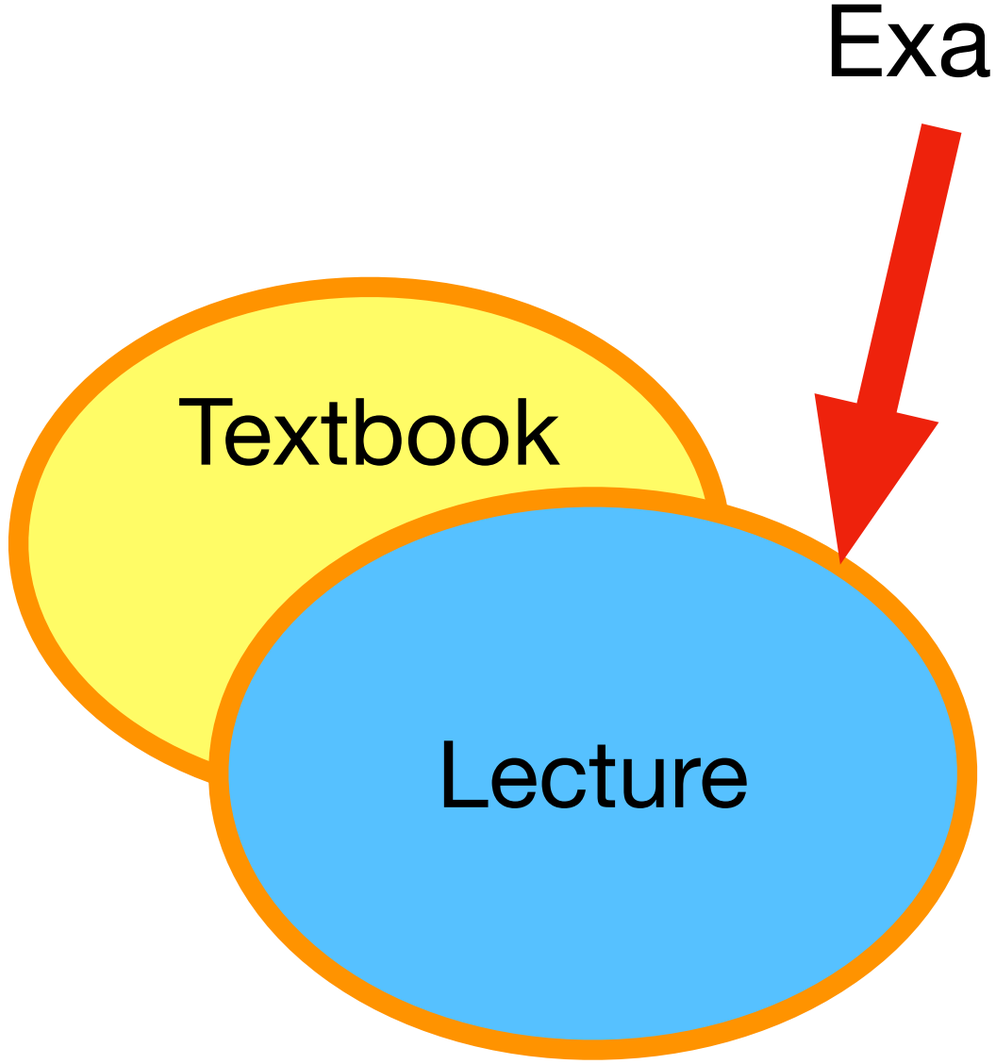
- Our course relates to Chapters 2-7

- Referred to “Epp” later

- Very helpful, though optional

- Suggestion: Skim the related chapter before the lecture; read deeper after the lecture

- Textbook may not cover everything in the exams



Overall schedule

- Lectures: Monday and Wednesday 3h30pm - 4h50pm, at C107
- Recitation: Tuesday 3h30pm - 4h25pm, at C107
- Office hours: Monday and Wednesday 2pm - 3pm, at B424/Zoom
- TA office hours: Tuesday, Thursday, Friday, from 8pm. Location and duration to be decided and discussed with TA.
- Per-class schedule is here: https://github.com/zhoulaifu/22_cse215_spring#schedule-tentative

Per-class schedule

Week	Date	Tentative schedule for each class	Reading
01	02-21	Overview of the course	
	02-22	Recitation: Look and feel of a final exam	
	02-23	Propositional logic [HW1 announced]	Epp, Ch2
02	02-28	Propositional logic	Epp, Ch2
	No class 03-01		
	03-02	Propositional logic	Epp, Ch2
03	03-07	Predicate logic [HW1 due; HW2 announced]	Epp, Ch3

- Slides will be available by the end of each lecture day

- Homework is due by the end of the due date, and announced by the end of the announcement day

- Per-class schedule might change. Important changes will be announced at Blackboard as early as is possible.

Homework

- Seven pieces of homework corresponding to seven chapters. They are denoted as HW1, HW2,… HW7 in per-class schedule.

- TA will take care of homework grading. By default, BlackBoard will be used for you to submit homework and for TA to grade.

Exams and grading

- We will use two midterm exams for self-evaluation. In particular, the 2nd midterm exam will take place near the semester end. So it can be see as a warm up for the final.

- Numerical Grading is a sum of
  - (HW1 + … + HW7) / 7 *** 30%**
  - (Midterm1 + Midterm2) / 2 *** 20%**
  - Final * **50%**

Please do not hesitate to ask questions

- CSE215 forum at blackboard for most Q & A. Perhaps subscribe to it so you get email notification.

- Private questions can be sent to TA or me, but this is not generally encourage as response time can be slower than the Forum.

- For urgency, feel free to call the instructor 010-4755-7557.

Questions so far?

More technic overview of the course

A personal story

The story

- Once upon a time, I worked for a project involving financial calculation
- I needed to sum up a number of decimal values like
  - 0.1 + 0.2 + 0.3 + 0.7 + 0.9 + 1.2 + 3.5…
- There were billion of numbers like this, so performance was a key for the project's success
- We decided to use the state-of-the-art multi-core, parallel computing
- Parallel computing works like a divide-and-conquer:
  - (0.1 + 0.2) + (0.3 + 0.7) + (0.9 + 1.2 + 3.5) + …
- Now, let us think why it looks reasonable le to use parallel computing for this task??
- The reason is associative law. (a + b) + c = a + (b + c)

The problem

We got different results for each round!

Why?

- We made this assumption:
  - for any numbers a, b, c, (a + b) + c = a + (b+ c)
- This is a statement that can be assigned with true or false value, we call it a **proposition**
- The inner part has variables, and can be denoted as a statement with parameters (a, b, c). We call it a **predicate**.
- Many CS work involves determining if a proposition is true or false. To show the truth is called **to prove**.
- The reason for the problem is that the proposition above is false.

Live demo:

First check 1 + 2 + 3, then 0.1 + 0.2 + 0.3

Summary for the story

The whole is a proposition, since we can assign a truth or false value to it

$$\forall a, b, c \in I, (a + b) + c = a + (b + c)$$

quantifier

variables

set

Predicate

A puzzle if time allows

- Assume associative law:

$$\forall a, b, c \in I, (a + b) + c = a + (b + c)$$

- Assume 1+1=2, 2+1=3

- Try to prove 1+2=3

Takeaway for today

- The course is for understanding why our digital world works or fails. It consists of fundamentals of the entire computer science and engineering.

- We will study logic (propositions, and predicates), proof, and math structures like sets

- Practical matter: covid protocol, schedule, ask questions