



# CSCI 101

## Connecting with Computer Science

### Lecture 1: Introduction



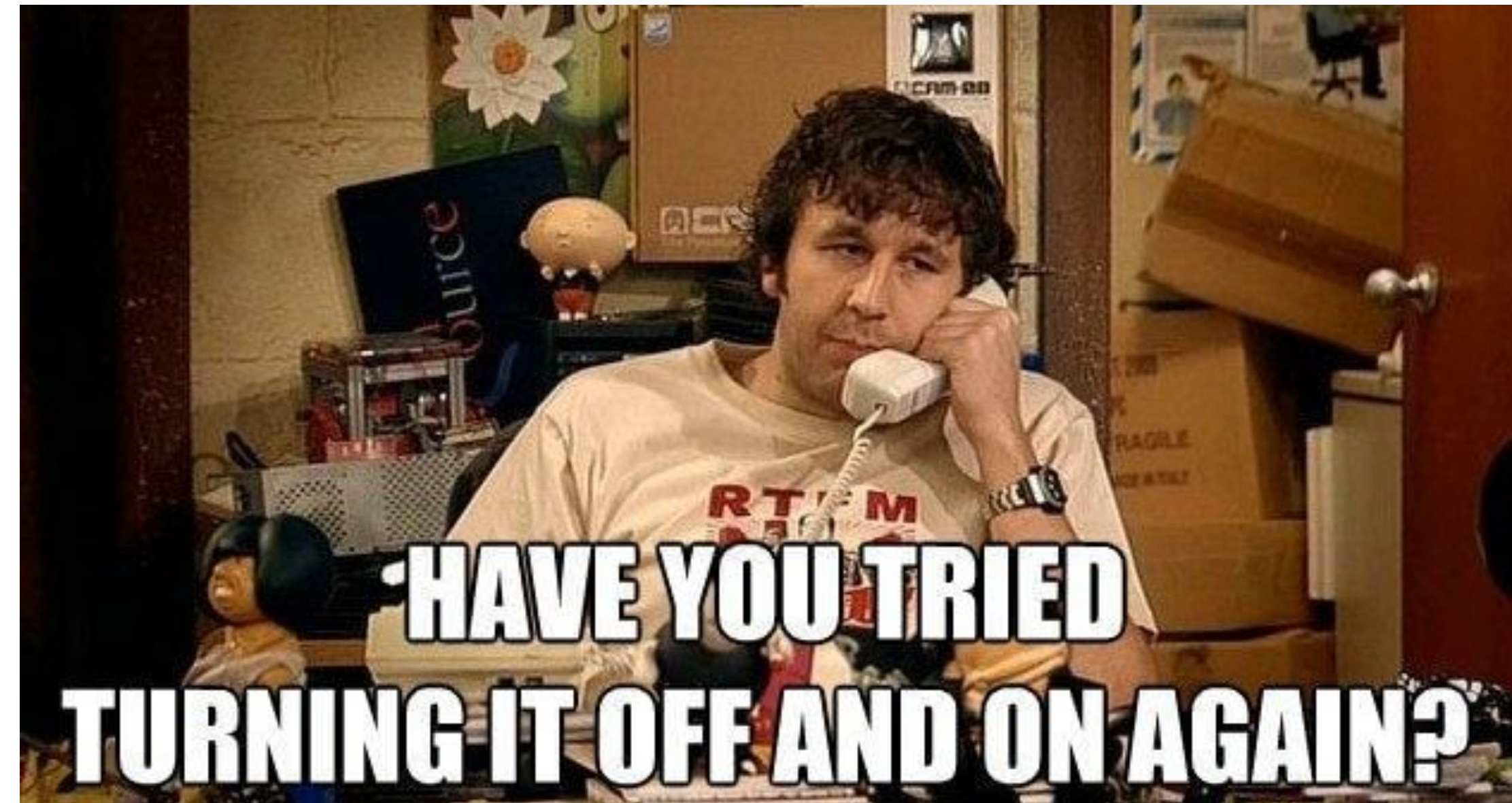
Jetic Gū  
2020 Fall Semester (S3)

# Overview

- Focus: Introduction to CS
- Architecture: Chat
- Core Ideas:
  1. What is Computer Science? What are the other stuff that you thought were CS?
  2. Roadmap to CS: A Systematic Overview

# What is Computer Science?

- IT: Information Technology
- Computer Engineering
- Software Engineering
- Data Science; Data Analysis
- Artificial Intelligence
- Cloud Computing



# What is Science?

- Discovering the way the physical world works (e.g. Laws of Gravity)
- Scientific
  - Verifiability: a theory can stand field tests
  - Consistency: the same test when repeated gives consistent results
- Empirical
  - Theory are supported by empirical experiment

# What is Engineering?

- Practical knowledge transforming scientific theories into products
- "Without engineers, science is just philosophy."

# Information Technology

- Equivalent to Computer Science
- Anything that has anything to do with information processing using non-human labour (basically, computing machinery)

# Computer Engineering

- Subset of CS/IT
- More about hardware design
- e.g. Embedded system, Networking equipment, Scientific/High-Performance Design, Multimedia hardware chips

# Software Engineering

- Subset of CS/IT
- Vast majority of IT professionals
  - Developing software required by customers/project managers
- Systematic design of complex software systems



# Data Science/Analysis

- Subset mostly Math (Statistics) and CS/IT
- NOT Science: there is no science of data
  - DA is more on the Engineering side
- Use computer as tools, try to statistically analyse the data
  - e.g. How many people visited [amazon.ca](https://www.amazon.ca) after Googling the term 'Refrigerator'?
  - e.g. "60% probability of 10% profit increase if we invest 100K on XXX advertising"

# Artificial Intelligence

- Subset of mostly Math (Statistics) and CS/IT
- Pure empirical: we don't have very good theories of why it works
  - or why it doesn't work: lack of verifiability

# Cloud Computing

- Subset of CS/IT
  - Stuff that you used to do on your own computer is now done by Amazon/Google/Microsoft online
  - You just see the results and interact with their servers
  - Engineering: a complicated hardware/software design problem
  - Science: optimisation, optimisation, optimisation

# Roadmap to CS

- You are (or considering) majoring in CS (B.Sc), now what?
- First two years of study: Basic of CS
  - Python, C/C++, Software Engineering Theories, OOP, Logical Circuits and Computer Organisation, algorithms and data structures, etc.
- Last 2 years of B.Sc: Pinpoint your interest, develop in-depth professional skills
  - SE, Cloud, Communications, Hardware, Graphics, AI, Network, Robotics, etc.

# Roadmap to CS

- Columbia College (or first 2 years)
  - CSCI 101: fun
  - CSCI 120, CSCI 125: Python, C/C++. Basic programming skills
  - CSCI 150, CSCI 250, CSCI 295: Computer Hardware. How Computers work
  - CSCI 165: Internet
  - CSCI 225: Data Structure and Programming: algorithms
  - CSCI 237: CS for business students
  - CSCI 275: Basic Software Engineering stuff

# Roadmap to CS

- Columbia College (or first 2 years)
  - CSCI 101: fun
  - CSCI 120, CSCI 125: Python, C/C++. Basic programming skills
  - CSCI 150, CSCI 250, CSCI 295: Computer Hardware. How Computers work
  - CSCI 165: Internet
  - CSCI 225: Data Structure and Programming: algorithms
  - CSCI 237: CS for business students
  - CSCI 275: Basic Software Engineering stuff

# Roadmap to CS

- University (or last 2 years of B.Sc)
- Advanced Algorithms, Operating System. Database System, Advanced Networks

