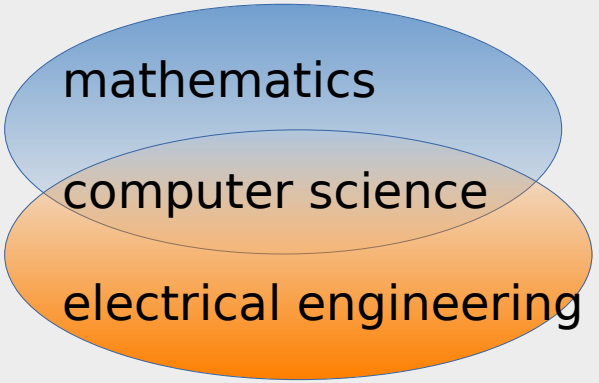
# ICS 111
# Introduction to computer science

- Origins
- Modern Computer Science
- Algorithms
- Systems
- Software Engineering
- Artificial Intelligence
- Data Science
- Human-Computer Interaction
- Security
- and much more!

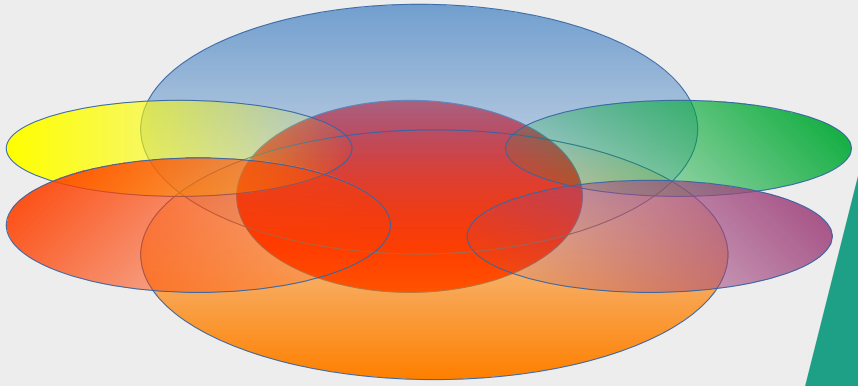
# ICS 111
# Origins of computer science

- Many early computer scientists came from Mathematics
  - they wanted to study and understand computation, create better algorithms
- Many were electrical (electronics) engineers
  - they wanted to build faster and more powerful machines
  - and storage, networks, displays, …
- Alan Turing was more of a mathematician, but involved in designing computers
- Konrad Zuse was more of an engineer interested in building computing machines
- John von Neumann was both



# ICS 111
# Modern computer science

- Overlapping with many, many fields
  - psychology and neuroscience (human-computer interaction)
  - biology (bioinformatics)
  - physics and astronomy
  - business
  - art, movies, games
  - robotics,
  - etc
- core includes:
  - algorithms and information theory
  - computer architecture and systems
  - software engineering
  - artificial intelligence, robotics, machine learning, computer vision, human-computer interaction
  - more recently, data science



# ICS 111
# Algorithms

- define a computation in the abstract, without reference to a specific machine – this is an **algorithm**
  - need some assumptions about what a machine can do in a single step, or in constant time
  - algorithm design is always an area of research
- mathematically analyze the performance, select the fastest algorithm
  - the fastest algorithm doesn’t always give the fastest program!
  - but usually does
  - especially as the amount of data grows
- originally this field focused mostly on sequential algorithms, now research often looks at concurrent and parallel algorithms

# ICS 111
# Systems

- computer architecture: instruction sets and their implementations
- databases: provide fast access to large amounts of structured data
- networks: protocols for machine-to-machine communication, the Internet
- for each of these, design, build, and analyze
- administration and management of each of these is part of Information Technology (IT), often (e.g. at UH) overlapping with business

# ICS 111
# Software Engineering

- programming language design
- compilers, IDEs, debuggers, repositories
- reusable code, APIs such as the Java standard library
- project design and implementation, from requirements to testing and maintenance
- managing complexity, delivering results
- why do programs break? analyzing bugs and security faults

# ICS 111
# Artificial Intelligence

- make computers do hard things, including understanding speech, and seeing like humans
  - ideally, some version of having the computer “think”
  - Turing test: can we create a program that talks in such a way that a human cannot distinguish between talking with a human vs. talking with a computer?
    - captchas address the reverse problem, where a computer has to make the distinction
- machine learning looks for patterns in real-world data
  - often adapting methods from biology (e.g. neural networks, evolution) or statistics
- robotics: building autonomous machines

# ICS 111
# Data Science

- computers are collecting more and more real-world data, leading to **big data**
- analyzing this data may give previously unexpected insights or may yield new information
- overlap with machine learning, and with many of the natural sciences
  - e.g. the search for the Higgs Boson at CERN was facilitated by analysis of massive amounts of data
  - DNA matching and analysis is a form of data science
- data science complements other approaches to science, often summed up as theoretical and empirical/experimental

# ICS 111
# Human-Computer Interaction

- making computers more user-friendly
- computer graphics and visualization
- includes technology:
  - immersive displays such as the Lava lab
  - 3D displays
  - interaction devices: mouse, tablet, gesture, emotion recognition
- overlap with psychology and brain science: what can we perceive?
- overlap with design and art

# ICS 111
# Security

- mathematical foundation of ciphers, encryption, and authentication
- modeling of actors and permissions (capabilities)
- practical concerns
  - a system designed to be secure, is likely more secure than adding security to an insecure system
  - off-line backups can help protect against ransomware
  - a system that is only secure if humans don’t make mistakes, is not secure!
- attacks (e.g. viruses, DoS) and defenses

# ICS 111
# … and much more!

- quantum computing
- games and interactive fiction
- medical informatics
- web programming and transactions
- embedded systems, including Internet of Things (IoT), Arduino, Raspberry Pi
- quantitative financial analysis