

# CS21: INTRODUCTION TO COMPUTER SCIENCE

---

Prof. Mathieson

Fall 2017

Swarthmore College

# Outline Sept 6:

Make sure to sign in again  
(for the waitlist)

- Introductions
- Nonograms followup
- Syllabus highlights
- Areas of computer science
  
- Intro to **python** and the interpreter
- Variables and assignment
- Types and conversion: **int**, **float**, **str**
- **input** function

# Nonogram followup

# Handout example

							2		
					2	2	1		
			1	5	1	3	1	5	1
		1							
		3							
	2	2							
	2	2							
		5							
1	1	1							
	1	3							

# Systematic solving of a nonogram

			2						
			2 2 1						
			1 5 1 3 1 5 1						
		1							
		3							
2	2		■				■		
2	2		■				■		
		5							
1	1	1							
		1 3							

# Systematic solving of a nonogram

							2			
					2	2	1			
			1	5	1	3	1	5	1	
		1								
		3								
	2	2	■				■			
	2	2	■				■			
		5	■				■			
1	1	1								
	1	3								

							2			
					2	2	1			
			1	5	1	3	1	5	1	
		1								
		3								
	2	2	■				■			
	2	2	■				■			
		5	■				■			
		×	×	■	■	■	■	×	×	
1	1	1								
	1	3								

# Systematic solving of a nonogram

							2		
					2	2	1		
			1	5	1	3	1	5	1
		1							
		3							
	2	2	■				■		
	2	2							
		5	■				■		
1	1	1							
	1	3							

							2		
					2	2	1		
			1	5	1	3	1	5	1
		1							
		3							
	2	2	■				■		
	2	2							
		5	■				■		
1	1	1	×	×	■	■	■	■	×
	1	3							

							2		
					2	2	1		
			1	5	1	3	1	5	1
		1							
		3							
	2	2	■			×	×	■	
	2	2							
		5	■			■	■	■	
1	1	1	×	×	■	■	■	■	×
	1	3							

# Solution

							2		
					2	2	1		
			1	5	1	3	1	5	1
		1							
		3							
	2	2							
	2	2							
		5							
1	1	1							
	1	3							

# Syllabus highlights

<https://www.cs.swarthmore.edu/courses/CS21Labs/f17/index.php?section=1>

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)
- Labs are (almost always) due Saturdays before midnight

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)
- Labs are (almost always) due Saturdays before midnight
- **Extensions**
  - Known conflicts: must be arranged **now**
  - Emergencies: must talk to your class dean

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)
- Labs are (almost always) due Saturdays before midnight
- **Extensions**
  - Known conflicts: must be arranged **now**
  - Emergencies: must talk to your class dean
- **Email** – allow 24 hours for a response

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)
- Labs are (almost always) due Saturdays before midnight
- **Extensions**
  - Known conflicts: must be arranged **now**
  - Emergencies: must talk to your class dean
- **Email** – allow 24 hours for a response
- **Piazza** should be used for all content/logistics questions

# Syllabus highlights and notes

- Notes will be be posted *after* class on the webpage
- Textbook – free and online 😊
- Make sure to attend lab! (mandatory, attendance taken)
- Labs are (almost always) due Saturdays before midnight
- **Extensions**
  - Known conflicts: must be arranged **now**
  - Emergencies: must talk to your class dean
- **Email** – allow 24 hours for a response
- **Piazza** should be used for all content/logistics questions
- Office hours: **3-5pm on Fridays in 260** (often moved to lab)

# Class Deans contact info

CLASS	DEAN	To SCHEDULE AN APPOINTMENT WITH YOUR DEAN
First-Year	Dean Karen Henry	Betsy Durning 610-690-5744 <a href="mailto:edurnin1@swarthmore.edu">edurnin1@swarthmore.edu</a>
Sophomore	Dean Jason Rivera	Stephanie Holznagel (assists with schedule only) 610-690-3999 <a href="mailto:sholzna1@swarthmore.edu">sholzna1@swarthmore.edu</a>
Junior	Dean Dion Lewis	Bonnie Lytle 610-328-8456 <a href="mailto:dlytle1@swarthmore.edu">dlytle1@swarthmore.edu</a>
Senior	Dean Nathan Miller	Stephanie Holznagel 610-690-3999 <a href="mailto:sholzna1@swarthmore.edu">sholzna1@swarthmore.edu</a>

# Disability Services

<http://www.swarthmore.edu/academic-advising-support/welcome-to-student-disability-service>

## Registering with the Student Disability Service

Please contact Leslie Hempling, Director of Student Disability Services, at [lhempli1@swarthmore.edu](mailto:lhempli1@swarthmore.edu) or 610-690-5014 to arrange an intake appointment. We are happy to hold initial appointments for incoming students by phone. If at all possible, please submit documentation of your disability in advance so that we can review it prior to talking with you. We recommend that you contact us as early as possible since some accommodations (e.g., electronic books, interpreters, etc.) can take a several weeks to arrange. We want to be sure that your needs are met in time for classes.

Visit the Accommodations Process and the Documentation Guidelines sections in the "[For Students](#)" section of this website for all details.

# Areas of Computer Science

# Computer Science Research Areas

- Artificial Intelligence
- Computer Architecture
- Computational Biology
- Databases
- Computer Science Education
- Computer Graphics
- Human-Computer Interaction
- Operating Systems
- Programming Languages
- Scientific Computing
- Cyber Security
- Theory

# Artificial Intelligence

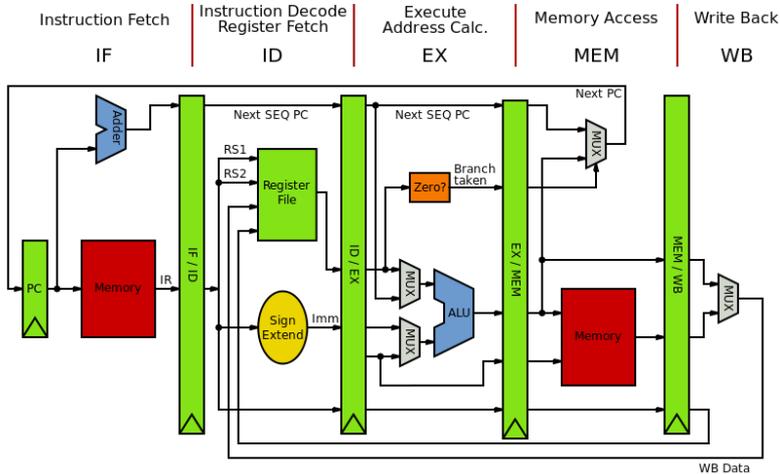


Jeremy Hsu, IEEE Spectrum

# Artificial Intelligence



Jeremy Hsu, IEEE Spectrum



Credit: InductiveLoad on Wikipedia

# Computer Architecture

# Artificial Intelligence



Jeremy Hsu, IEEE Spectrum

# Computational Biology

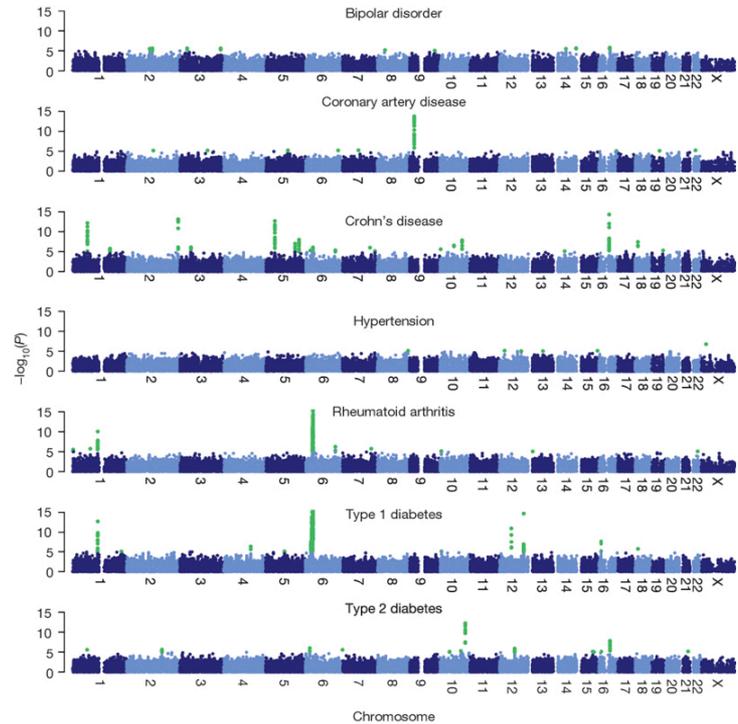
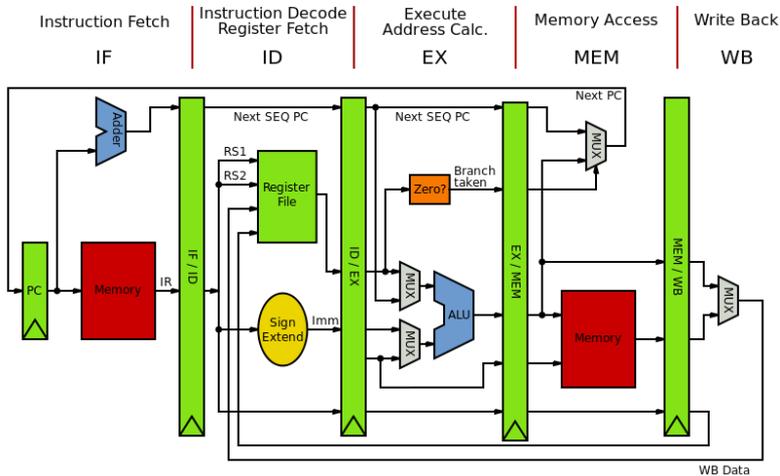


Figure credit: "Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls" by the Wellcome Trust Case Control Consortium

Credit: InductiveLoad on Wikipedia



# Computer Architecture

# Databases

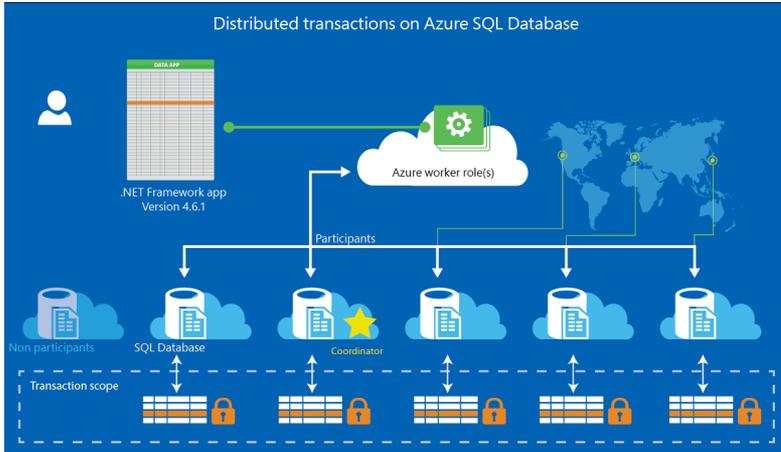


Figure credit: "Distributed transactions across cloud databases"  
– Microsoft Azure Database

# Databases

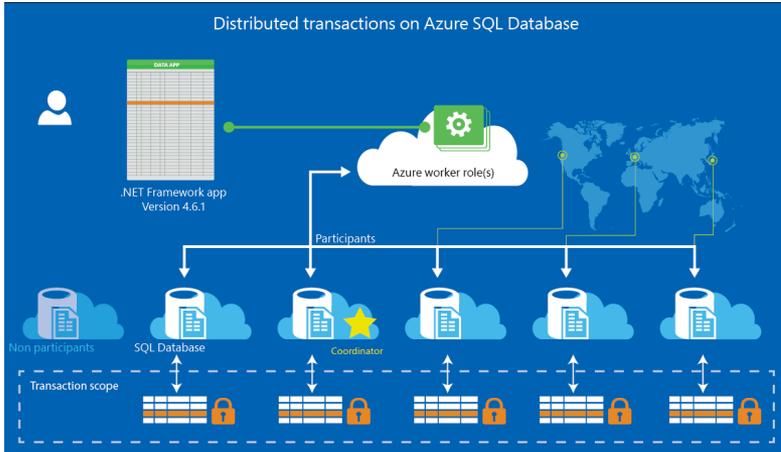


Figure credit: "Distributed transactions across cloud databases" – Microsoft Azure Database



Scratch Programming - EngagingEducators.com

# Computer Science Education

# Databases

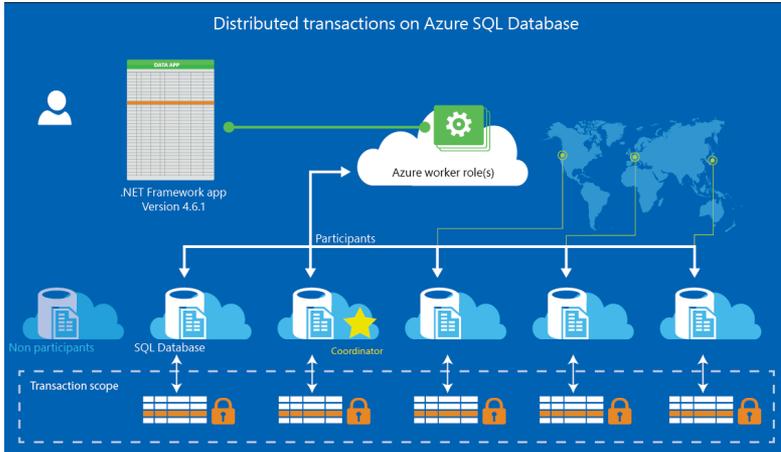


Figure credit: "Distributed transactions across cloud databases"  
– Microsoft Azure Database

# Computer Graphics



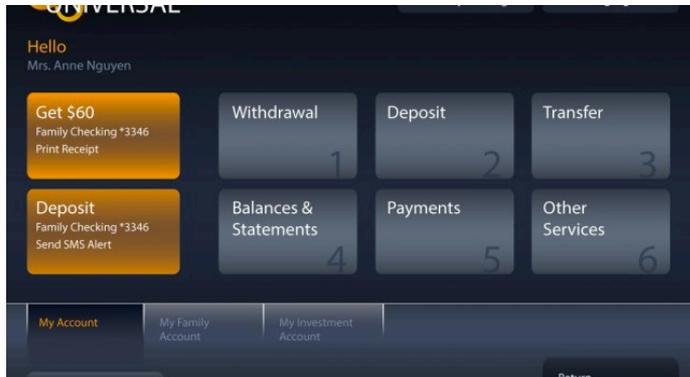
Scratch Programming - EngagingEducators.com



"Zootopia"  
Disney, 2016

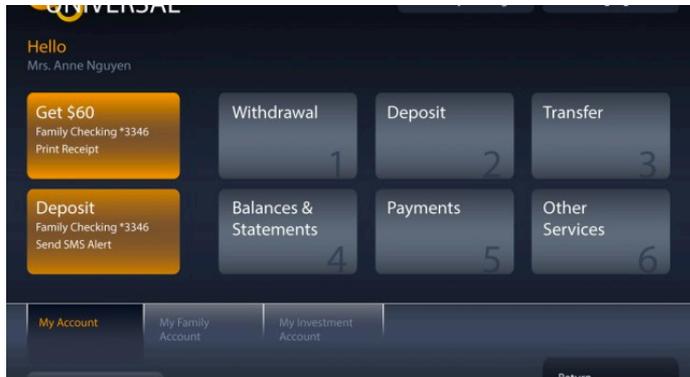
# Computer Science Education

# Human-Computer Interaction

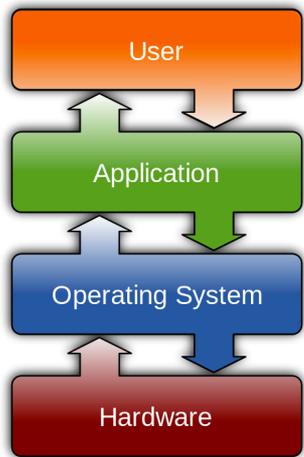


Universal ATM Interface by Kristin Suzanne Bessette

# Human-Computer Interaction



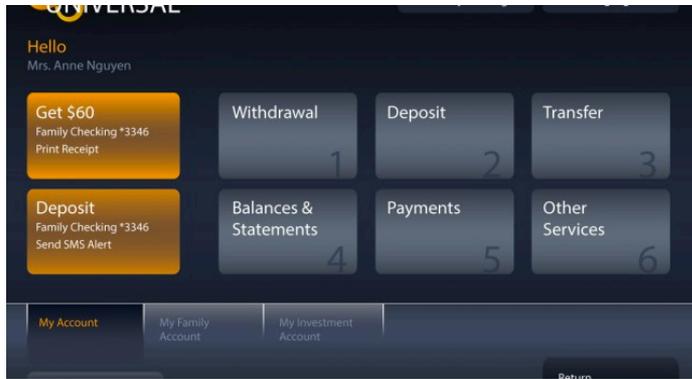
Universal ATM Interface by Kristin Suzanne Bessette



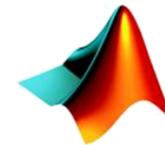
Wikimedia Commons (by Golftheman)

## Operating Systems

# Human-Computer Interaction



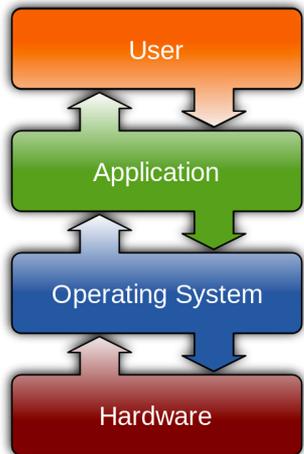
## Programming Languages



MATLAB



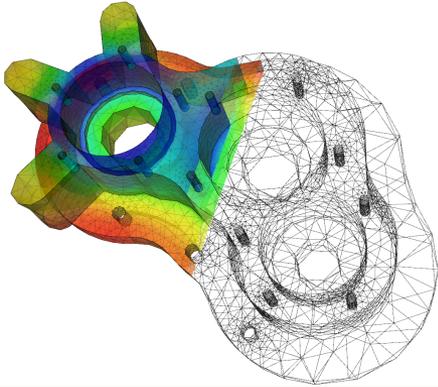
Universal ATM Interface by Kristin Suzanne Bessette



Wikimedia Commons (by Golftheman)

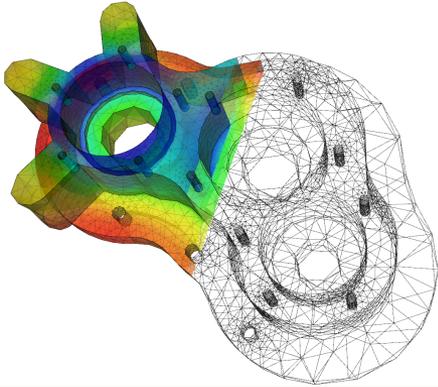
## Operating Systems

# Scientific Computing



"A numerical solution to the heat equation on a pump casing model using the finite element method." – Wikipedia

# Scientific Computing



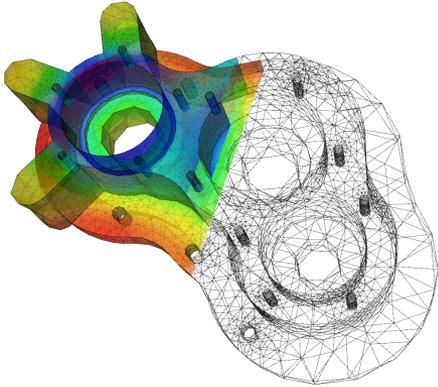
"A numerical solution to the heat equation on a pump casing model using the finite element method." – Wikipedia



Credit: Hugh Boyes (2014)

# Cyber Security

# Scientific Computing



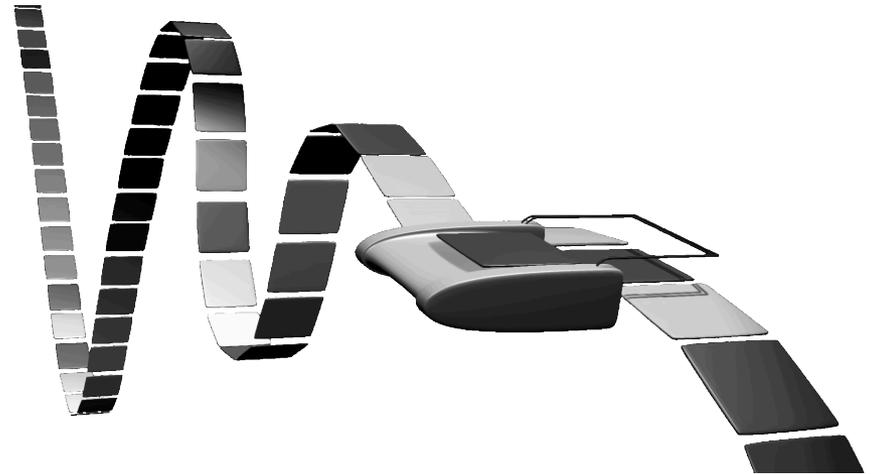
"A numerical solution to the heat equation on a pump casing model using the finite element method." – Wikipedia



# Cyber Security

Credit: Hugh Boyes (2014)

# Theory



"An artistic representation of a Turing machine. Turing machines are used to model general computing devices." – Wikipedia

# Python interpreter: demo

# Key CS concepts today

- *Variables* as a way to store *values*
- *Assignment operator (=)* is a way to change the value of a variable (not symmetric like equals operator in math!)
  - Variable name on the left, expression on the right
- The *type* of a variable is the type of the value it refers to
- We can *convert* a variable to a different type, but it does not change the value of the original variable

# Functions for today

- `type()`
- `input()`
- `int()`
- `str()`
- `float()`

# Types for the first few weeks

- `int`
- `float`
- `str`