CS4HS 2017
Why Computer Science?
Why UW CSE?

Ed Lazowska
Bill & Melinda Gates Chair
Paul G. Allen School of Computer Science & Engineering
University of Washington

July 2017
Why Computer Science?

Why UW CSE?

Ed Lazowska

Bill & Melinda Gates Chair
Paul G. Allen School of Computer Science & Engineering
University of Washington

July 2017
Forty five years ago ... 

Credit: Peter Lee, Microsoft Research
THE ARPA NETWORK
DEC 1969
4 NODES

29 OCT 69 2100
LOADED OP. PROGRAM \( SK \)
FOR BEN BARKER
BBV

22:30 TALKED TO SRI
HOST TO HOST

LEFT IMP. PROGRAM \( SL \)
RUNNING AFTER SENDING
A HOST DEAD MESSAGE
TO IMP.
With 4+ decades of hindsight, which had the greatest impact?

• Unless you’re big into Tang and Velcro (or sex and drugs), the answer is clear ...

• And so is the reason ...

EXPONENTIAL US
Exponentials are rare – we’re not used to them, so they catch us unaware.
Every aspect of computing has experienced exponential improvement

- Processing capacity
- Storage capacity
- Network bandwidth
- Sensors
- Astonishingly, even algorithms in some cases!
You can exploit these improvements in two ways

- Constant capability at exponentially decreasing cost
- Exponentially increasing capability at constant cost
The 1970s to today

1970 Ford Mustang

2014 Ford Mustang

Size: roughly comparable
Speed: roughly comparable
Efficiency (MPG): roughly comparable
Value (cost relative to performance): roughly comparable
1971 Intel 4004
(2,300 transistors)

2014 Intel Xeon
(4,300,000,000 transistors)

Size: area occupied by a transistor reduced by 1,000,000x
Speed: operations per second increased by 100,000x
Efficiency (operations per watt): improved by 6,750x
Value (dollars per instruction): improved by 2,700x
What if cars had improved as rapidly as microprocessors?
The 1970s to today

Size: A car would be smaller than an ant!
   (About 1/5\textsuperscript{th} of an inch long!)
The 1970s to today

Speed: A car would go 6,000,000 miles per hour! (San Francisco to New York in 1.7 seconds!)
The 1970s to today

Efficiency: A car would get 100,000 miles per gallon!
(San Francisco to New York on 1/2 cup of fuel!)
The 1970s to today

Cost: A car would cost less than $10!
During the decade of the 2000’s ...

- Search
- Scalability
- Digital media
- Mobility
- eCommerce
- The Cloud
- Social networking and crowd-sourcing
During the current decade ...

- Smart homes
- Smart cars
- Smart health
- Smart robots
- Smart crowds and human-computer systems
- Smart education
- Smart interaction (virtual and augmented reality)
- Smart cities
- Smart discovery
Smart homes (the leaf nodes of the smart grid)

Shwetak Patel,
University of Washington
2011 MacArthur Fellow
Smart cars

DARPA Grand Challenge

DARPA Urban Challenge

Google Self-Driving Car

Tesla Model S

Adaptive cruise control

Self-parking
Smart health

Larry Smarr – “quantified self”

Evidence-based medicine

P4 medicine, “scientific wellness”
Smart robots
Smart crowds and human-computer systems

Zoran Popovic,
UW Computer Science & Engineering

David Baker,
UW Biochemistry
Smart education

Zoran Popovic,
UW Computer Science & Engineering
Smart interaction
Smart cities
Smart discovery (data-intensive discovery, or eScience)

Nearly every field of discovery is transitioning from “data poor” to “data rich”
A 21st century view of Computer Science:
A field that’s unique in its societal impact

CORE CSE
AI, systems, theory, languages, etc.

- mobile computing
- natural language processing
- sensors
- data science
- cloud computing
- robotics
- computer vision
- human computer interaction
- machine learning

- Energy & Sustainability
- Medicine & Global Health
- Transportation
- Education
- Scientific Discovery
- Neural Engineering
- Elder Care
- Accessibility
- Technology Policy and Societal Implications
- Security, Privacy, & Safety
- Advancing the Developing World
- Interacting with the Physical World: “The Internet of Things”
Computer Science: The ever-expanding sphere
Students are figuring this out!

- Demand is booming at colleges and universities nationwide
  - For introductory courses
  - For the major
  - For upper-division and graduate courses by non-majors
Demand for introductory courses: Students are realizing that every 21st century citizen needs to have facility with “computational thinking” – problem analysis and decomposition (stepwise refinement), abstraction, algorithmic thinking, algorithmic expression, stepwise fault isolation (debugging), modeling

- Computational thinking is not “this particular operating system” or “that particular programming language.”
- Computational thinking is not even programming. It’s a mode of thought – a way of approaching the world.
- Programming is the hands-on, inquiry-based way that we teach computational thinking and the principles of computer science.
• Demand for upper-division and graduate courses by non-majors: Students are realizing that computer science is great preparation for anything! Fields from Anthropology to Zoology are becoming *information* fields, and that those who can bend the power of the computer to their will – computational thinking, but also computer science in greater depth – will be positioned for greater success than those who can’t
  — Data science is a perfect example
• Demand for the major: Students are realizing that computer science is not Dilbert – it’s an intellectually exciting, highly creative and interactive, “power to change the world” field
• Students are also realizing that pretty much all of the STEM jobs are in computer science
  – While fluency with computational thinking and with computer science are important to all fields, the job prospects in the field of computer science itself are extraordinary
    • The U.S. Bureau of Labor Statistics recently released its job projections for the decade 2014-2024. Computer occupations will be responsible for 73% of all the job growth in all fields of STEM (Science, Technology, Engineering, and Mathematics) – the many dozens of fields that comprise the life sciences, the physical sciences, the social sciences, engineering, and the mathematical sciences – and for 55% of all available jobs, whether newly-created or available due to replacement
    • In Washington State, the workforce gap in computer science is greater than the workforce gap in all other fields (not just STEM fields!) combined
Job Growth, 2014-24

10%
6%
5%
3%
3%

73%

U.S. Bureau of Labor Statistics

- Computer occupations (15-1100)
- Engineers (17-2000)
- Life scientists (19-1000)
- Physical scientists (19-2000)
- Social scientists and related workers (19-3000)
- Mathematical science occupations (15-2000)

Job Openings (Growth+Replacement), 2014-24

55%

Data from the spreadsheet at http://www.bls.gov/emp/ind-occ-matrix/occupation.xlsx
High-Demand Fields in Washington State, Baccalaureate Level
(Washington Student Achievement Council / State Board for Community & Technical Colleges / Workforce Training & Education Coordinating Board, 2016)

- **Computer Science**
- **Engineering**
- **Human & Protective Services**
- **Media, Design & Communications***

*Driven by the technology sector

- Annual Degree Completers Entering the Workforce
- Additional Annual Completions Needed, 2018-2023
Every high school should offer computer science, and every student should take it!

• *Not* because programming is a valuable skill (although it certainly is that)
• Rather, *because every field is becoming an information field*
• And *because “computational thinking” is an essential 21st century capability*
Is this a great time or what?