Philosophical Issues of Computer Science
Computing between science and engineering
Instructor: Viola Schiaffonati
March, 31st 2016
THE PHILOSOPHICAL ISSUES OF CS

Viewpoint
The Science in Computer Science

“The question of “scienceness” of computing has always been complicated because of the strong presence of science, mathematics, and engineering in the roots and practice of the field. [...] Computing is now accepted as science. Some of us even believe computing is so pervasive that it qualifies as a new domain of science alongside the traditional domains of physical, life, and social sciences.” (Denning 2013)

“To the degree that some aspects of computing are subject to analysis and modeling, it is fair to say that there is a rigorous element of science in our field.” (Cerf 2012)
Reopening the debate

- Computing’s *disciplinary identity* (today)
  - Is an all-inclusive definition of computing as a discipline necessary?
  - Does computing stand out as an example of a post-disciplinary era of science?
  - Does computing represent a new kind of science?
  - ...

Philosophical Issues of CS
A paradigmatic case

- The case of "experimental computer science" is emblematic under many respects
  - Calling for experiments in computing as a way to assess its scientific status
  - Naïve notion of experiment in many cases
  - Full adequacy to the same standards of traditional experimental sciences
“Computer science is an empirical discipline. We would have called it an experimental science, but like astronomy, economics, and geology, some of its unique forms of observation and experience do not fit a narrow stereotype of the experimental method. None the less, they are experiments. Each new machine that is built is an experiment. Actually constructing the machine poses a question to nature; and we listen for the answer by observing the machine in operation and analyzing it by all analytical and measurement means available. Each new program that is built is an experiment. It poses a question to nature, and its behavior offers clues to an answer. Neither machines nor programs are black boxes; they are artifacts that have been designed, both hardware and software, and we can open them up and look inside. We can relate their structure to their behavior and draw many lessons from a single experiment.” (Newell and Simon 1976)
"Let us employ traditional measures when assessing experimental computer science. Let us always have a clear plan for testing a clear hypothesis. Let us not call "hacking" science. These are the criteria by which the rest of the world will evaluate our field's experimental work. If we do not live up to the traditional standards of science, there will come a time when no one takes us seriously."

(Denning 1980)
“Experimentation is central to the scientific process. Only experiments test theories. Only experiments can explore critical factors and bring new phenomena to light so that theories can be formulated and corrected. Without experiments, computer science is in danger of drying up and becoming an auxiliary discipline. The current pressure to concentrate on application is the writing on the wall. I don’t doubt that computer science is a fundamental science of great intellectual depth and importance. Much has already been achieved. Computer technology has changed society, and computer science is in the process of deeply affecting the world view of the general public. There is also much evidence suggesting that the scientific method does apply. As computer science leaves adolescence behind, I hope to see the experimental branch of this discipline flourish.”

(Tichy 1998)
These examples and other extant computer science theories emphasize that by embracing the methodology of developing and evaluating predictive models through experimentation over multiple members of a class of software systems, a more complete understanding of such artifacts will emerge. [...] How can these benefits be realized? How might we change what we do? We can adapt our already very skilled hypothesis testing in debugging and broaden it by asking more general questions [...] The pristine presentations of scientific reasoning and the tremendous successes of such reasoning in other fields may appear to the practicing computer scientist as out of reach. But many of our colleagues have started down this path, the tools are accessible, and the promise is great.”

(Morrison and Snodgrass 2011)
Computing as science (Denning and Rosenbloom 2009)

- Satisfiability of 6 criteria
  - Directed and systematic knowledge
  - Reproducible results
  - Well developed experimental methods
  - Abilities of prediction
  - Ability to produce falsifiable hypotheses
  - Dealing with natural objects
On the scientific status of computing

- Computing as the study of *information processes*, both *natural* and *artificial* ones
- Computer: from object of study to tool for studying
  - Computational science
- Computing is not computational science
  - Use of the computers to provide detailed insights into the behavior of complex physical system
Computing: a new paradigm

- Beyond the contraposition between science and engineering
- Widening the framework (Internet, Web science, mobile computing, user interface design, information visualization, ...)
- Increasing adoption of experimental methods to understand computations
  - Design and discovery approach: behavior of large systems too complex to be analyzed in pure mathematical terms and requires to be discovered by means of rigorous observations
- Computing as the fourth great domain of science
  - Computing is fundamental as the physical, life, and social sciences
Computing as a separate domain

- It has a **distinctive focus**: computation and information processes
- Its constituent fields and its structures and processes are in constant **interaction**: computer science, informatics, information technology, computer engineering, software engineering, and information systems
- Its **influence** is extensive and pervasive, reaching deep into people’s lives and work
Is computer science an engineering discipline?

- Argument relying on the view that the goal of computer science is to design and construct **useful things**
- Unlike natural scientists who deal with natural occurring phenomena engineers deal with artifacts which are created by people
“Engineering as a profession is identified with the **systematic knowledge** of how to design **useful artifacts** or **processes**, a discipline that includes some **pure science** and **mathematics**, the “applied” or “engineering sciences” [...] and is directed toward some **need or desire**. But while engineering involves a relationship to these other elements, **artifact design** is what constitutes the essence of engineering, because it is design that establishes and orders the **unique engineering framework** that integrates other elements”

(Mitcham 1994)
What is engineering?

"The creative application of scientific principles to design or develop structures, machines, apparatus, or manufacturing processes, or works utilising them singly or in combination; or to construct or operate the same with full cognizance of their design; or to forecast their behaviour under specific operating conditions; all as respects an intended function, economics of operation and safety to life and property"

(Vincenti 1990)
Computer scientist as toolsmith

“The scientist builds in order to study, and the engineer studies in order to build”

(Brooks 1996)
Computer science as a synthetic discipline

- Computer science is not a proper science
- **Unhappy trends** in computer science
  - Theory more respected than practice
  - Regarding as an end the invention and publication of endless varieties of computers, algorithms, and languages
  - Forgetting the users and their real problems
  - Directing young and brilliant minds toward theoretical subjects
Driving-problem approach

- The approach relies on working on the problems of another discipline: to focus on another field can help
  - To focus on relevant problems, not just on exercises or at toy-scale problems
  - To be honest about success and failure
  - To face the whole problem (not just the easy or mathematical parts)
  - To learn or develop new computer science
  - To have fun
Computer science is a new engineering discipline

Goal of engineering: production of useful things, economically, for the benefit of the society

These things can be artifacts, but also manufacturing and chemical processes

A process is similar to a computational algorithm which is a method for processing information

Computer science is concerned with producing useful things given economic constraints requiring to seek efficient solutions to technical problems (like traditional engineering disciplines)
Engineering disciplines have their scientific roots in the *engineering sciences* (statics, dynamics, mechanics of solids, fluid mechanics, ...)

The *scientific fundamentals* of computer science include: Boolean algebra, computability theory, automata and formal languages

Fundamental concepts and principles of computer science are rooted not in the physical phenomena of force, heat, or electricity, but in *mathematics*

Computer science is therefore a new kind of engineering
Critique to the traditional conventional distinction between computer science (just theory) and computer engineering (everything else)

Excluding theory of computation from computer engineering is inconceivable

Theories provide effective models for the design, analysis, and understanding of programs and circuits

In their daily work engineers do not apply their discipline’s theory directly, but frequently rely in well established heuristics
References