

Lecture 12

Review

Computer and Logic Essentials



Semester 1 2020

Today

- 1 Computational thinking
- 2 History
- 3 Programmers
- 4 Unit review
- 5 Exam
- 6 From here

What is computational thinking?

- ▶ Designing computations that get computers to do jobs for us
- ▶ Explaining and interpreting the world as a complex of information processes

Denning and Tedre, Computational Thinking

CT is not a set of concepts for programming ... CT comprises ways of thinking and practicing that are sharpened and honed through practice.

Denning and Tedre, Computational Thinking

Wishful thinking

Can we get computers to do anything we want?

For example, there is no algorithm that will inspect another algorithm and tell us whether it terminates or loops forever.

There are some things we can do using a computer, but not in a reasonable time.

Denning and Tedre, Computational Thinking

More wishful thinking

Does knowing how to program enable you to solve problems in any field that uses computation?

You will need to know something about the other field as well.

Denning and Tedre, Computational Thinking

And some more

Do you believe computers are smart?
If you do not precisely specify your steps, your computation will contain errors that could cause disasters.

Denning and Tedre, Computational Thinking

Mathematical methods

Euclid’s algorithm from 300 BCE is an example of a mathematical method.

Sieve of Erastothenes

This method finds prime numbers by eliminating multiples.

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
31 32 33 34 35 ...

Computational machines

In around 1650, machines were built to perform repetitive calculations. These made use of wheels and gears and contained representations for loops and conditions.

An example is Blaise Pascal's calculator which could add and subtract.

Human computers

In the mid-1900s the role of performing calculations was done by human computers.

They also used CT:

- ▶ breaking computation into pieces that can be done by separate, interacting computers
- ▶ arranging the computers to optimise communication and messaging
- ▶ including error checks

Automatic computers

The first electronic computer was the ENIAC, built by the US Army in the mid-40s; programs were set up by wiring up boards which took several days. The first stored program computer was the Manchester Baby, built in the late 1940s.

Before and after

In 1937 Claude Shannon demonstrated the use of Boolean algebra for designing circuits. Binary was also adopted for computing. In the 1940s, computing was established as an academic discipline and more modern computers were being built.

Programming languages

The first programming languages surfaced around this time, starting with machine and assembly languages, before high-level programming languages appeared in the 1950s.

FORTRAN (1954) is probably the most famous of these early languages still being used.

Pascal and C were developed in the late 1960s, around the time that the term “software engineering” was first used.

Who are programmers?

Originally female-dominated, however the field slowly became more male-dominated throughout the 60s/70s as programmers were considered to be skilled professionals.

Student enrolments shifted in the early 1980s with the introduction of home PCs; male students were more likely to start uni with programming experience.

Thompson, Coders

What programmers do

- ▶ Write code from scratch
- ▶ Fix bugs in code
- ▶ Fix bugs in other programmers' code
- ▶ Try to understand how humans interact with their software

Broader implications

- ▶ Provision of good UIs for users
- ▶ Social implications of software
- ▶ Financial considerations

Lecture 12

Review

Computer and Logic Essentials



Semester 1 2020