

MY470 Computer Programming

What Is Computation?

Week 1 Lecture, MT 2017

Overview

- Computational thinking and algorithms
- Computers, programming languages, and computer programs
- Objects, expressions, and variables

Computational Thinking

Computational Thinking is the thought processes involved in formulating a problem and expressing its solution in a way that a computer – human or machine – can effectively carry out.

Wing, Jeannette M. (2006). Computational thinking (<http://tech-insider.org/academia/research/acrobat/0603.pdf>). *Communications of the ACM*, 49(3), 33-35.

Defining Characteristics of Computational Thinking

Wing, Jeannette M. (2006). Computational thinking (<http://tech-insider.org/academia/research/acrobat/0603.pdf>). *Communications of the ACM*, 49(3), 33-35.

- **Conceptualizing**, not programming — requires thinking at multiple levels of abstraction
- A way that **humans**, not computers, think — requires cleverness and imagination
- Combines **mathematical and engineering** thinking — dictated by the constraints of physical computing devices
- For **everyone**, everywhere — just like reading, writing, and arithmetic

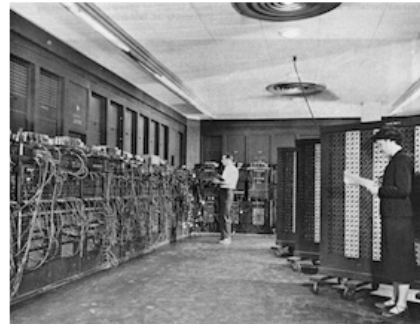
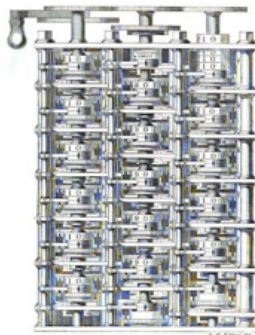
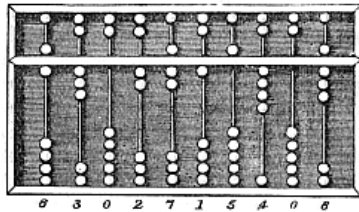
Algorithms

An algorithm is a well-defined computational procedure that takes value(s) as input and produces value(s) as output.

- "Recipe" or "instructions" for solving a well-defined computational problem
- Consists of a sequence of simple steps, control flow, and a stopping rule
- Can be specified in human language or programming language (or even as hardware design)

Computers

Computers automatically perform calculations, either built-in or user-defined, and store the results.



(Source: Wikimedia)

Programming Languages

A programming language is a formal language used to specify a set of instructions for a computer to execute.

- Primitive constructs
- Syntax
- Static semantics
- Semantics

Primitive Constructs in Programming Languages

- Literals

In [1]: 470

Out[1]: 470

In [2]: 'MY'

Out[2]: 'MY'

- Infix operators

In [3]: 470 / 3

Out[3]: 156.66666666666666

Syntax in Programming Languages

- Rules for forming strings of characters and symbols
- Programming languages have strict syntax

```
In [4]: 470 + 0.5
```

```
Out[4]: 470.5
```

```
In [5]: 470 0.5
```

```
File "<ipython-input-5-5a5b76bbe317>", line 1
```

```
470 0.5
```

```
^
```

```
SyntaxError: invalid syntax
```

Static Semantics in Programming Languages

- Rules for forming meaningful syntactically valid strings

In [1]: 'MY'/470

```
-----  
TypeError                                Traceback (most recent call last)  
<ipython-input-1-fb25aaf6edea> in <module>()  
----> 1 'MY'/470
```

```
TypeError: unsupported operand type(s) for /: 'str' and 'int'
```

Semantics in Programming Languages

- The meaning associated with a syntactically correct string that has no static semantic errors
- Programming languages have simple semantics — statements have only one meaning
- **But this may not be the meaning the programmer had in mind!**

Types of Programming Languages

- Low-level vs. high-level
- General vs. application-targetted
- Interpreted vs. compiled

Computer Program

- A sequence of definitions and commands
 - Commands (or "statements") instruct the computer to do something
- For interpreted languages:
 - Programs are executed by the language interpreter (or "shell")
 - They can be typed directly in the shell
 - Or they can be stored in a file and run from the shell

Objects, Data Types, and Expressions

- Programs manipulate objects
- Objects have types
 - Scalar – indivisible
 - Non-scalar – with internal structure
- Expressions combine objects and operators

```
In [ ]: # scalar objects
2
0.125
True

# non-scalar objects
'This is a string.'
[1, 2, 3, 'a', 'x']

# expressions
2 / 0.125
'MY' + '470'
```

Variables

- Variables associate objects with a name

```
In [ ]: a = 3.14  
        b = 11.2  
        c = a*(b**2)
```

```
In [ ]: pi = 3.14  
        diameter = 11.2  
        area = pi*(diameter**2)
```

- Variable names help humans read programs!
- Comments also improve readability!

```
In [ ]: pi = 3.14  
        diameter = 11.2 # diameter of circle  
        area = pi*((diameter/2)**2) # estimate area of circle using diameter
```

Computer Bugs

9/9


0800 Andam started
 1000 " stopped - andam ✓

1300 (032) MP-MC { 1.2700 9.037847025
 1.582647000 9.037846995 correct
 2.130476415 (032) 4.615925059(-2)
 (033) PRO 2 2.130476415
 correct 2.130676415

Relays 6-2 in 033 failed special speed test
 in relay 10,000 test.

Relays changed

1100 Started Cosine Tape (Sine check)
 1525 Started Multi-Adder Test.

1545  Relay #70 Panel F
 (moth) in relay.

First actual case of bug being found.

1630 Andam started.
 1700 closed down.

Relay 3371

The actual first computer bug. On September 9, 1947, Admiral Grace Hopper found this moth trapped on a relay of the Harvard Mark II computer. (Source: U.S. Naval Historical Center Online Library)

What Really is Computer Programming?

99 little bugs in the code,

99 bugs in the code,

1 bug fixed...run again,

100 little bugs in the code...

What Is Computation?

We use programming languages to write programs that instruct computers to perform algorithms, which calculate results or process data.

-
- **Lab:** Installing Anaconda, working with Jupyter, and uploading assignments on Github
 - **Next week:** Data types in Python