

Lecture Notes for Lecture 1 of CS 5001
(Intensive Introduction to Computer Science)
for the Fall, 2018 session at the Northeastern
University Silicon Valley Campus.

*Introduction to Computers, Programs, and
Programming Languages*

Philip Gust,
Clinical Instructor
Department of Computer Science

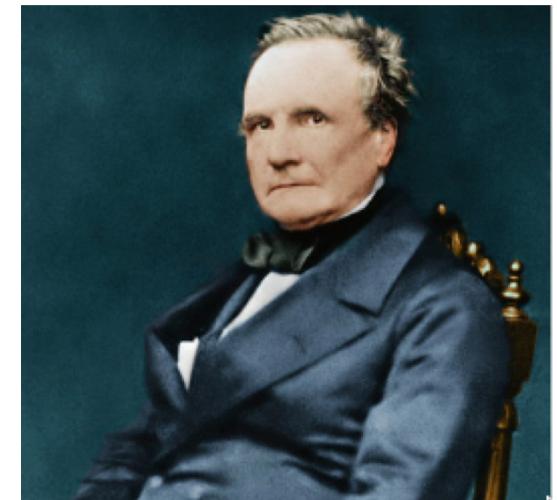
<http://www.ccis.northeastern.edu/home/pgust>

What is a Computer?

- A computer is a system that follows instructions to perform operations on input data, and produces output data to solve a problem.
 - A special-purpose computer has a fixed set of instructions that can solve a specific problem or class of problems.
 - A general-purpose computer can be programmed using a variable set of instructions to solve any kind of problem.
- A computer does not need to be electronic to be a computer: it could be mechanical or even biological.

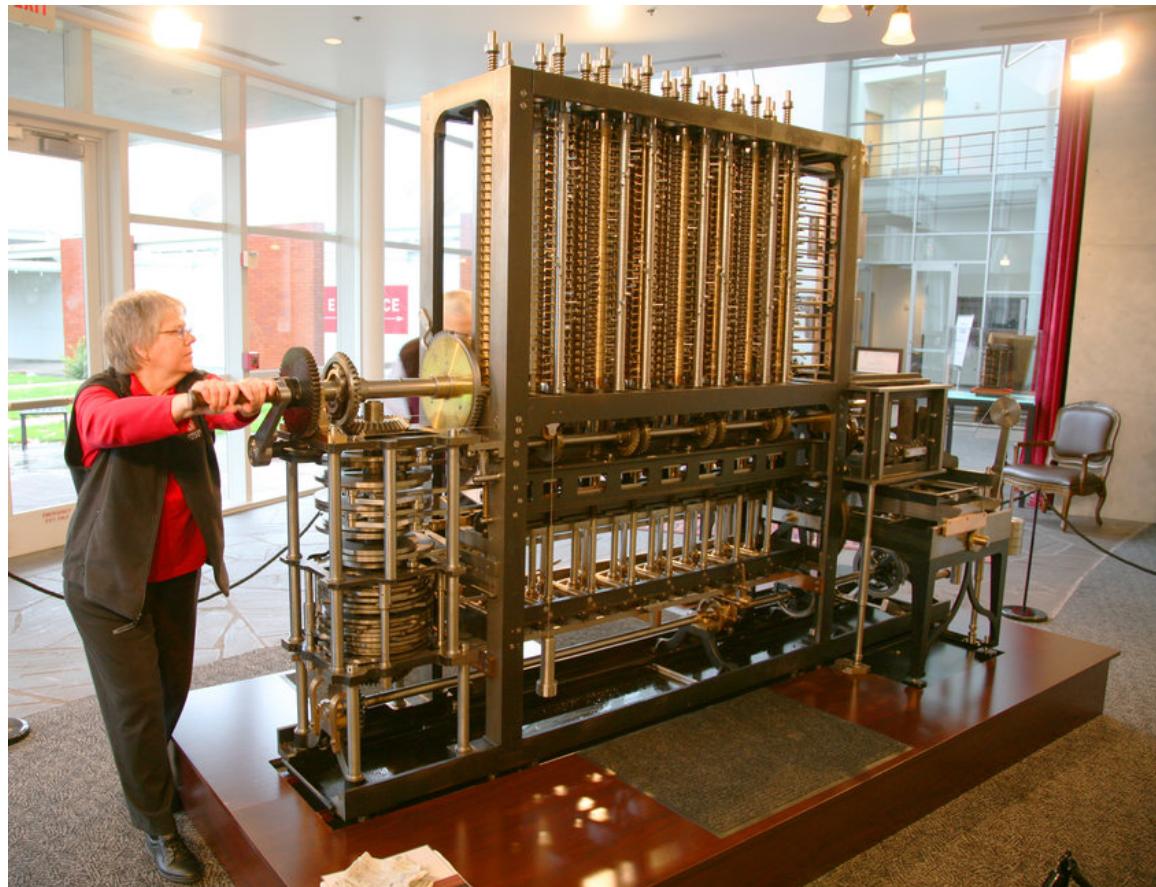
What is a Computer?

- English inventor Charles Babbage designed the Difference Engine in the early- to mid-1800s.
- It was, a special-purpose mechanical computer to automate producing mathematical tables.
- He was unable to complete it in his lifetime because parts could not be machined to required tolerances.
- Several were successfully built and operated in the early 2000s, including one displayed at the Computer History Museum in Mt. View, California.



What is a Computer?

- Reproduction of Babbage's Difference Engine #2



<https://www.youtube.com/watch?v=be1EM3gQkAY>

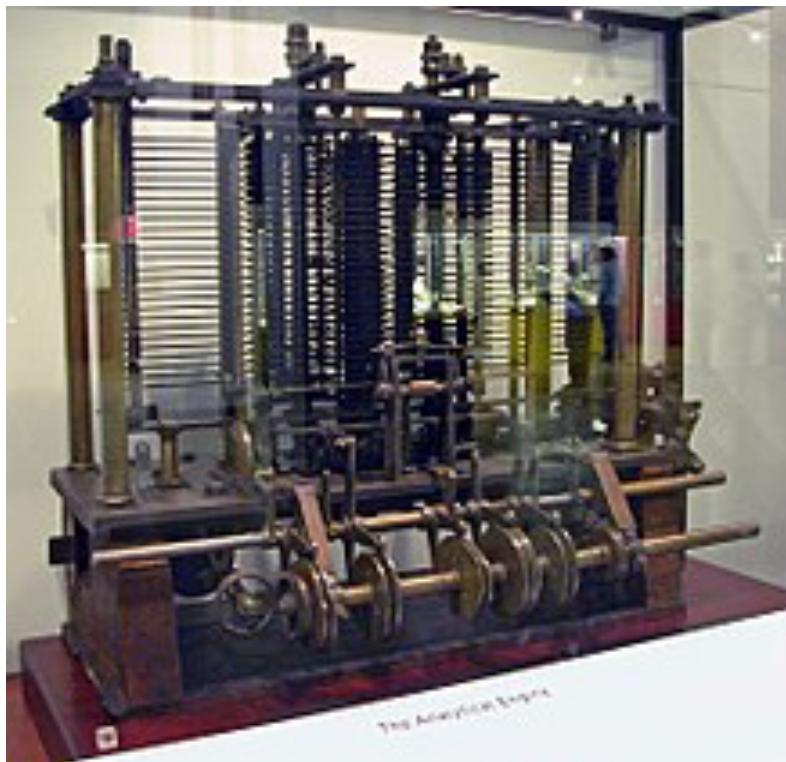
What is a Computer?

- In the 1840s, Babbage designed a general-purpose mechanical computer called the *Analytical Engine*.
- It would be fully programmable using punch cards based on automated looms of the time. There were cards for instructions and cards for data.
- Only parts of the machine have ever been built. It would have been slow by today's standards. Babbage estimated that it could

“...form the product of two numbers, each containing twenty figures, in three minutes.”

What is a Computer?

Trial model of part of Analytical Engine
built in Babbage's lifetime.



Punch cards for Analytical Engine:
operational (instruction) cards (foreground)
and variable (data) cards (background)

What is a Computer?

- English mathematician Ada Lovelace, a protégé of Babbage, wrote algorithms for the machine in the mid-1800s that most believe are the first computer programs.
- She also developed a vision of the capability of computers that went beyond mere calculating, including music and literature.
- She also considered the impact of technology like the Analytic Engine on society.



What is a Computer?

- In 1943, John Mauchly and J. Presper Eckert created the first electronic, general-purpose computer, ENIAC.
- It was built with vacuum tubes and was initially used to test the feasibility of nuclear weapons.
- Programming required changing plugs and switch positions, often requiring a week. Card programmable computers were a decade away.



What is a Computer?

- By late 1960s and early 1970s, special-purpose and general-purpose computers made their way into space



Apollo Guidance Computer, 1967
(special-purpose computer)



Voyager Onboard Computer, 1973
(general-purpose computer)

What is a Computer?



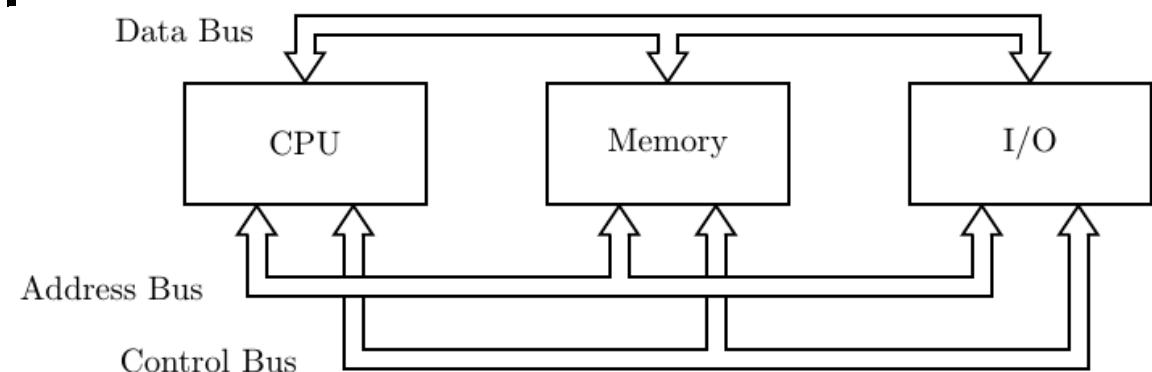
Tesla S with Nvidia Drive PX 2 Autocruise, 2017
(special-purpose computer)



Apple Macbook Pro with Intel I7, 2017
(general-purpose computer)

What is a Computer?

- The principal parts of a modern computer include:
 - Central processing unit (CPU)
 - Memory unit (MU)
 - Input/output unit (IOU)
- The units are connected by a data, address, and control bus.



What is a Computer?

- Computer memories represent information using electronic switches that are either off (0) or on (1): this is known as a *bit* or binary digit.
- Memory is organized into groups of 8 bits that are called a *byte*.
- A byte in memory can be read from or written to using its *address*.

w	
0	10110011
1	00000000
2	00000001
3	10000110
4	00111001
5	00000010
6	00011001
7	11111111
8	00100010
9	10101010
...	...
n-2	00000000
n-1	00000000

What is a Computer?

- A byte can be used to represent various types of data.
 - One byte can be used to represent up to 256 *characters*
 - A *character encoding* assigns a value to each character. A popular encoding, ASCII (American Standard Code for Information Interchange), assigns:
 - ‘0’ = 00110000 = 48_{10} = 60_8 = 30_{16}
 - ‘A’ = 01000001 = 65_{10} = 101_8 = 41_{16}
 - ‘a’ = 01100001 = 97_{10} = 141_8 = 61_{16}

What is a Computer?

- A byte can be used to represent various types of data.
 - A byte can also be used to represent an *unsigned number* between 0 (00000000) and 255 (11111111)

1	0	1	1	0	1	1	0
7	6	5	4	3	2	1	0

$$182_{10} = 266_8 = B6_{16}$$

- A byte can also be used to represent a signed number between -128 and 127 using the top bit as the *sign bit*.

sign bit ->

1	0	1	1	0	1	1	0
7	6	5	4	3	2	1	0

$$-54_{10} \text{ (sign-magnitude)}$$

sign bit ->

1	0	1	1	0	1	1	0
7	6	5	4	3	2	1	0

$$-73_{10} \text{ (1s-complement)}$$

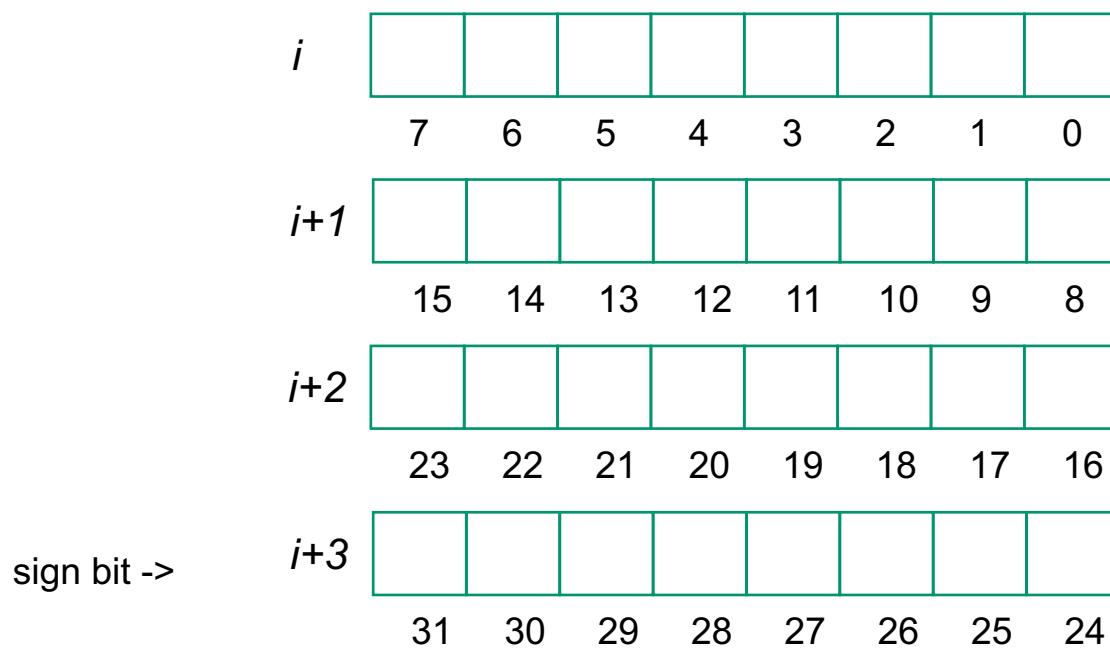
sign bit ->

1	0	1	1	0	1	1	0
7	6	5	4	3	2	1	0

$$-74_{10} \text{ (2s-complement)}$$

What is a Computer?

- Multiple bytes can be combined to represent larger values
 - An *integer* combines four sequential bytes that allow unsigned numbers between 0 and 4294967295_{10} , or signed numbers between -2147483648_{10} and $+2147483647_{10}$.

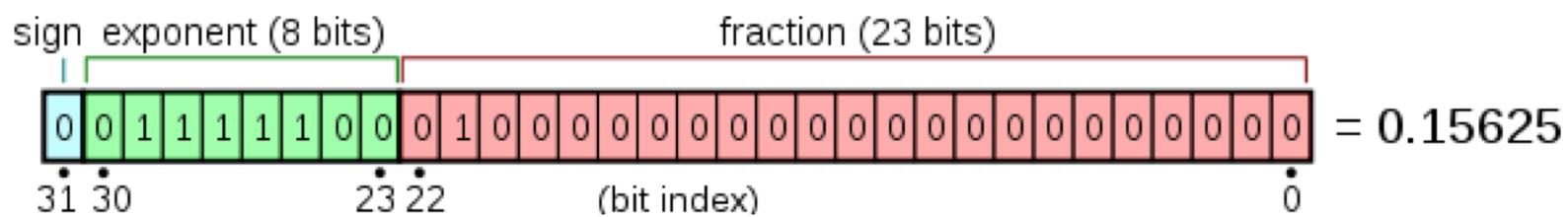


What is a Computer?

- Multiple bytes can be combined to represent larger values:
 - An *memory address* combines eight sequential bytes that allow addressing memory locations between 0 and $2^{64}-1$ (18446744073709551615).
 - A memory address is the address of the first byte of a value in memory. If a value is made up of multiple bytes, the required number of bytes starting at that location are used.
 - A memory address can also be stored in memory, and used to access a value indirectly using that address. The address is said to “point to” the value starting at that location.

What is a Computer?

- Multiple bytes can be combined to represent larger values:
 - A *floating-point* number combines four sequential bytes. The bytes are divided into fields that represent the sign, an exponent, and a mantissa – like scientific notation.
 - A floating point number is only an *approximation* of a real valued number.
 - Here is a floating point number in IEEE floating point format:

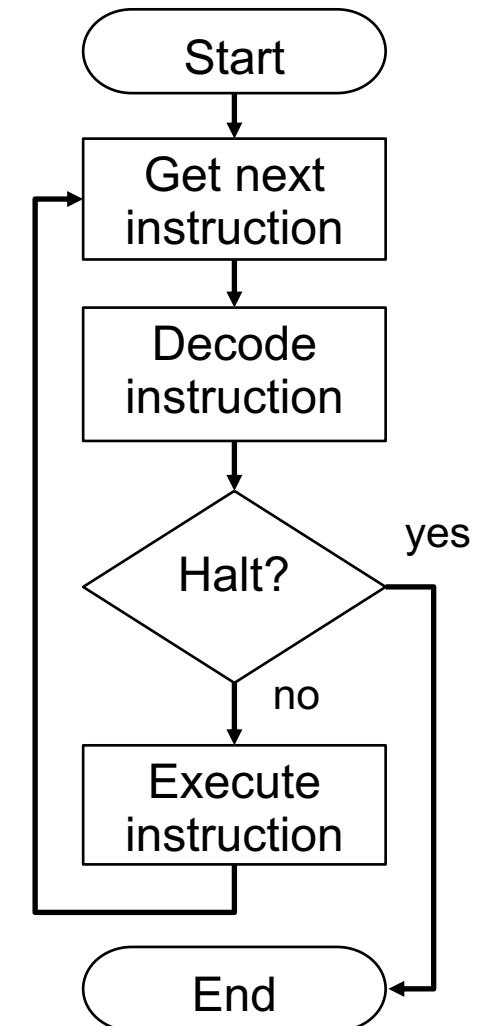


What is a Computer?

- Multiple bytes in memory can be combined to represent larger values:
 - The exponent (124_{10}) is “biased” by 127, so subtract 127 to get the actual value of -3. The binary point is before bit 22. The mantissa with sign uses two's-complement form.
 - See the IEEE floating point standard for more details on this format.
 - A floating point number can represent a range of $\pm 3.4E+38$. with the smallest positive value $1.2E-38$ and a precision of 6 digits.

What is a Computer?

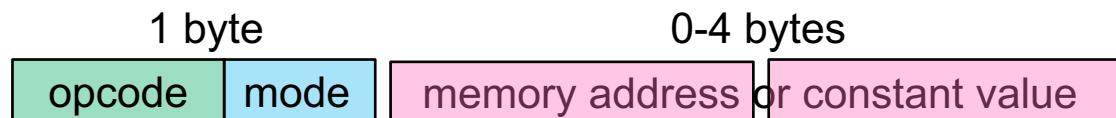
- The central processing unit (CPU) executes instructions and uses data fetched from memory.
- There are a fixed set of instructions. Each instruction performs a simple operations.
- Examples include arithmetic, boolean, decision, and I/O operations.
- The exact instructions and formats vary for each computer architecture.



Instruction Cycle

What is a Computer?

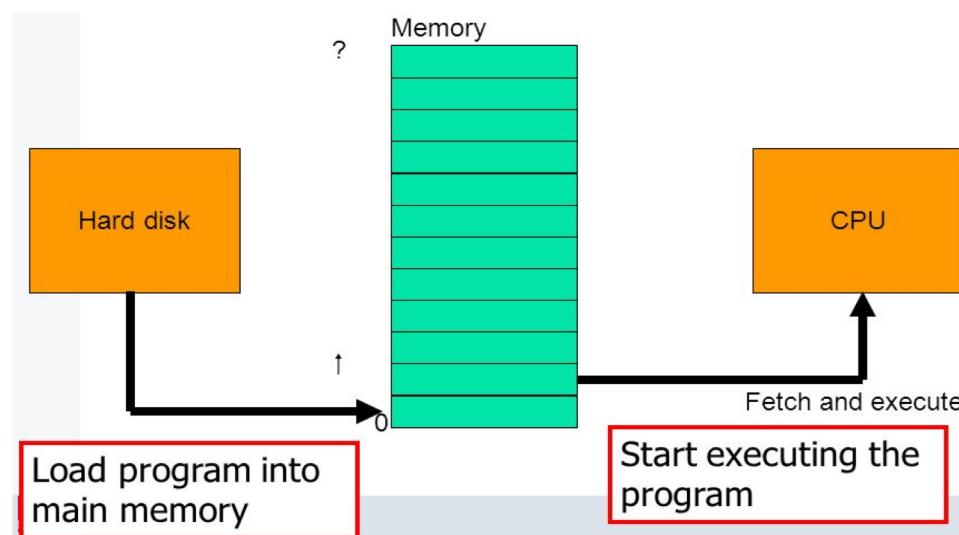
- An instruction has an *operand code (opcode)*, an *address mode*, and an address or data field encoded into one or more bytes stored in memory.
- Here is a typical instruction format. The number of bytes depends on the operation and address mode.



- Adding two numbers requires three instructions:
 - LOAD constant value or value from a memory location
 - ADD constant or value from another memory location
 - STORE result to a memory location

What is a Computer Program?

- A computer program is a sequence of instructions for performing some computation, and the data used for the computation
 - Data + Instructions = Program
- A program is loaded into memory, and execution starts at the address of the first instruction:



What is a Programming Language?

- A programming language is used to express an algorithm for solving a problem on a computer.
- The language has two components
 - instructions used to express an algorithm
 - data used by the instructions
- The programming language that includes the basic computer instructions and data is called *machine language* or *assembly language*.

Levels of Programming Languages

- **Low-level**
 - closely matches actual computer instructions and data types
- **Mid-level**
 - more complex instructions and data types
 - readily translated (compiled) to actual computer instructions and data types
 - requires minimal runtime support by programming language
- **High-level**
 - instructions and data types more closely match problem domain
 - no direct translation to actual computer instructions and data types
 - requires extensive runtime support by programming language

Language Paradigms

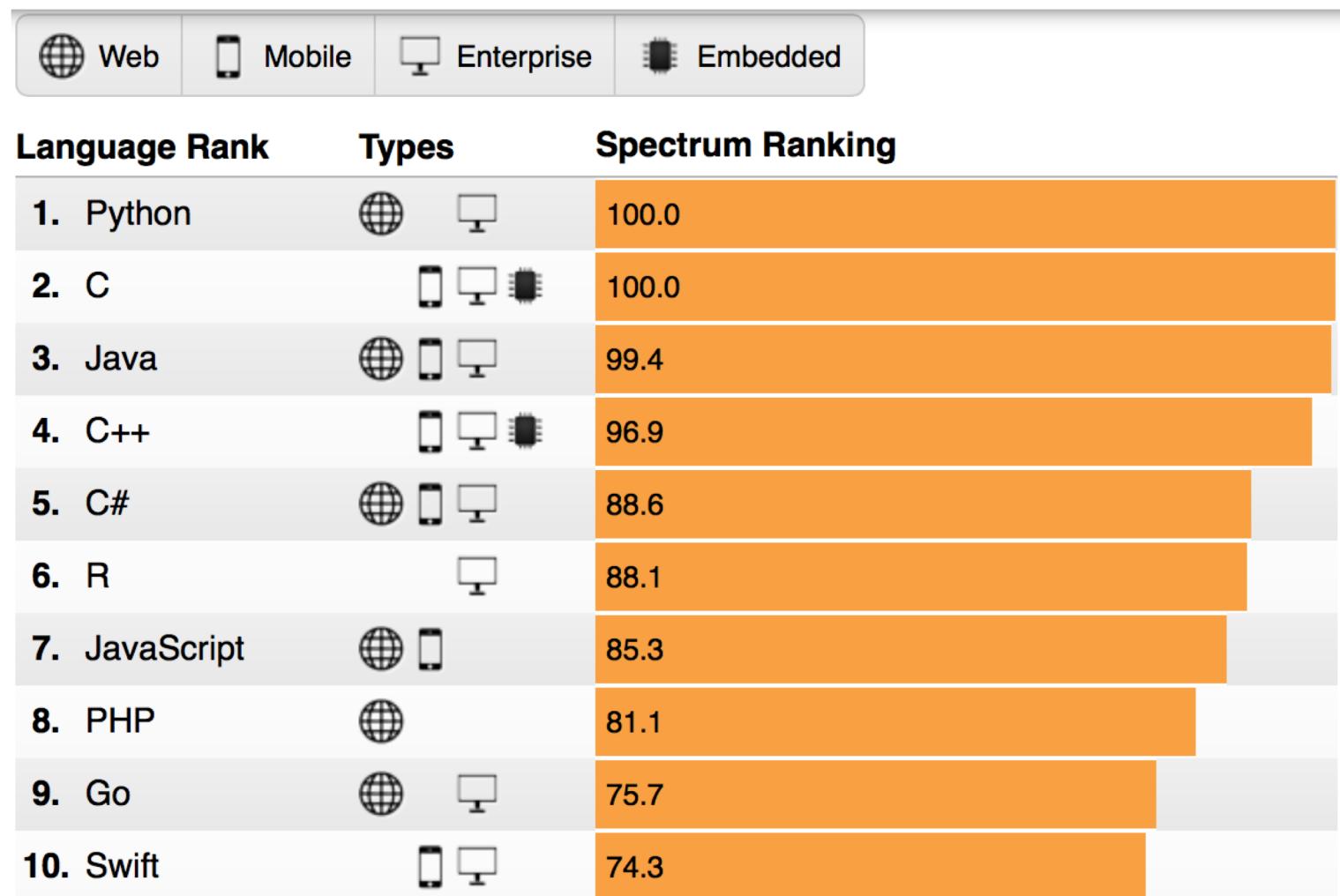
- **Imperative**
 - step-by-step instructions for solution
 - for each prescription bill received this month:
 - if the prescription bill amount is less than the allowable maximum:
 - add the bill to the list of deductible medical expenses
 - all computer instruction sets are imperative
- **Declarative**
 - expresses desired outcome, not how to achieve it:
 - only prescriptions bills less than the allowable maximum are deductible
- **Event-driven**
 - specifies what to do when a condition occurs in any order:
 - when a prescription is filled:
 - file as deductible if bill is less than the allowable maximum

Introduction to C

- We will studying C to learn how to solve problems using computer programs
 - C is a mid-level imperative programming language
 - easy to learn
 - can be translated into efficient program on any computer
- Fun facts about C:
 - Invented by Dennis Richie at AT&T Bell Labs to write Unix operating system
 - Named "C" as successor to earlier "B" programming language in the early 1970s
 - U.S. (American National Standards Organization), and international (International Standards Organization) standard language
 - Most state-of-art software is implemented in C
 - Top-ranked language for mobile, enterprise, and embedded software

Introduction to C

- Source: IEEE Spectrum survey 2017



Introduction to C

- C is a particular kind of imperative language known as "procedural"
 - A program is built from one or more procedures (also termed subroutines or functions):
 - procedure process medical bills:
 - for each prescription bill received this month:
 - classify the prescription bill
 - procedure classify a prescription bill:
 - if prescription bill is under the allowable maximum:
 - add bill to list of deductible medical expenses

Introduction to C

- C is a foundational language for computer science.
- It strongly influenced many subsequent, widely-used languages:
 - C++ (“C with objects”)
 - Java
 - C#
 - JavaScript / ECMAScript

Introduction to C

- A C program must be translated into machine instructions for a particular computer. The program that does this is called a *compiler*.
- The C compiler reads one or more text files that contain a C program. If there are no syntax errors, the compiler outputs translated machine instruction files.
- The C compiler combines these files with libraries for things like math and input/output, and links them into an executable binary program to run.

Introduction to C

- Process of compiling C program in files into binary machine instructions.

