Introduction to Scientific Computing I

Lecture 6

Amir Farbin

A Brief History of Programming

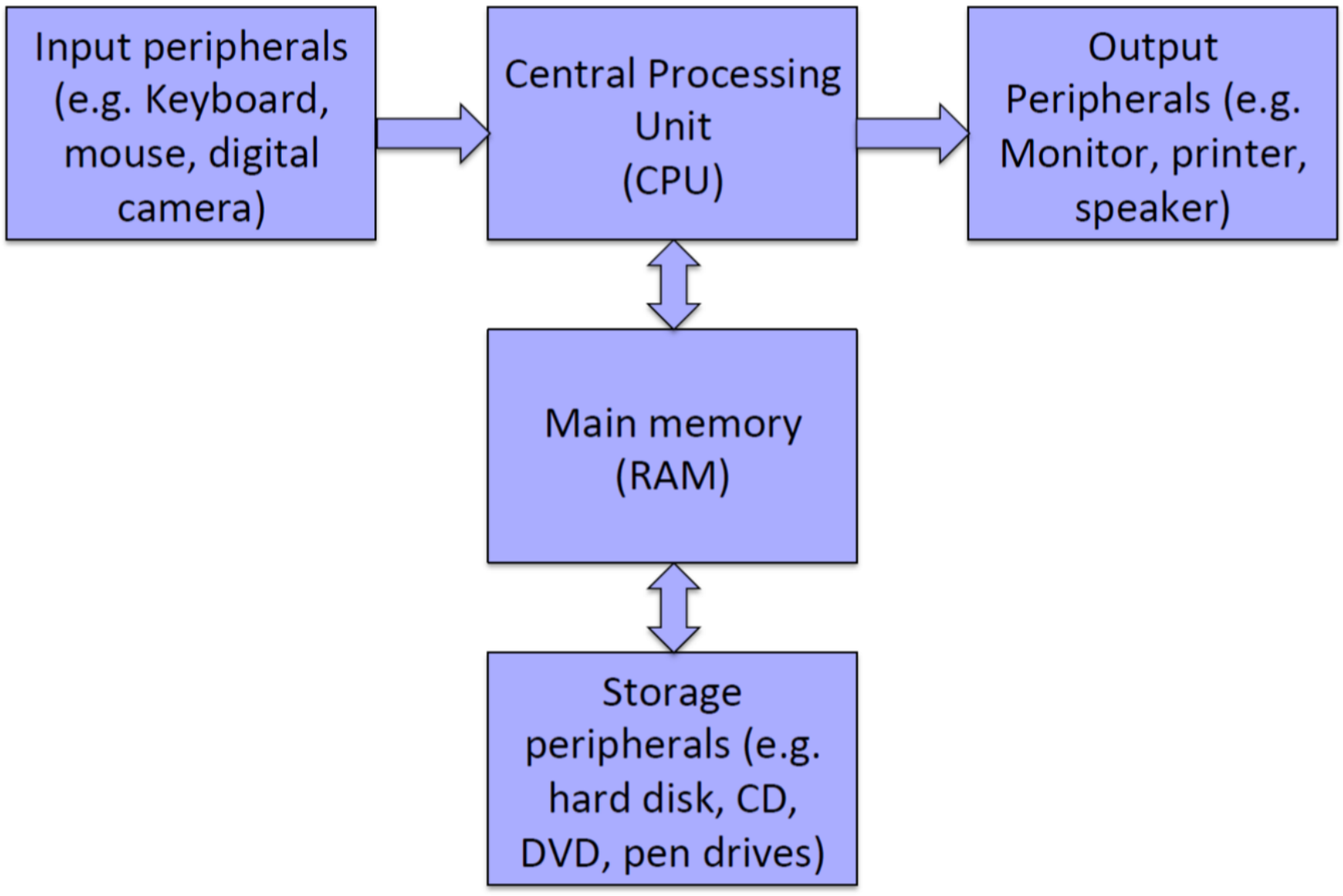
The rise of the machines

- Rapid growth in science and engineering after WW-I and during WW-II.
- Two fields with rapid growth
    - mathematical and physical sciences
        - mathematical programming (a terminology used commonly in place of mathematical optimization) (not to be confused with computer programming!) and Monte Carlo
- Computer Science (a new field of science and technology) began to rise during WW-II
    - response to the needs of war
    - response to the exponential growth of natural sciences and engineering.

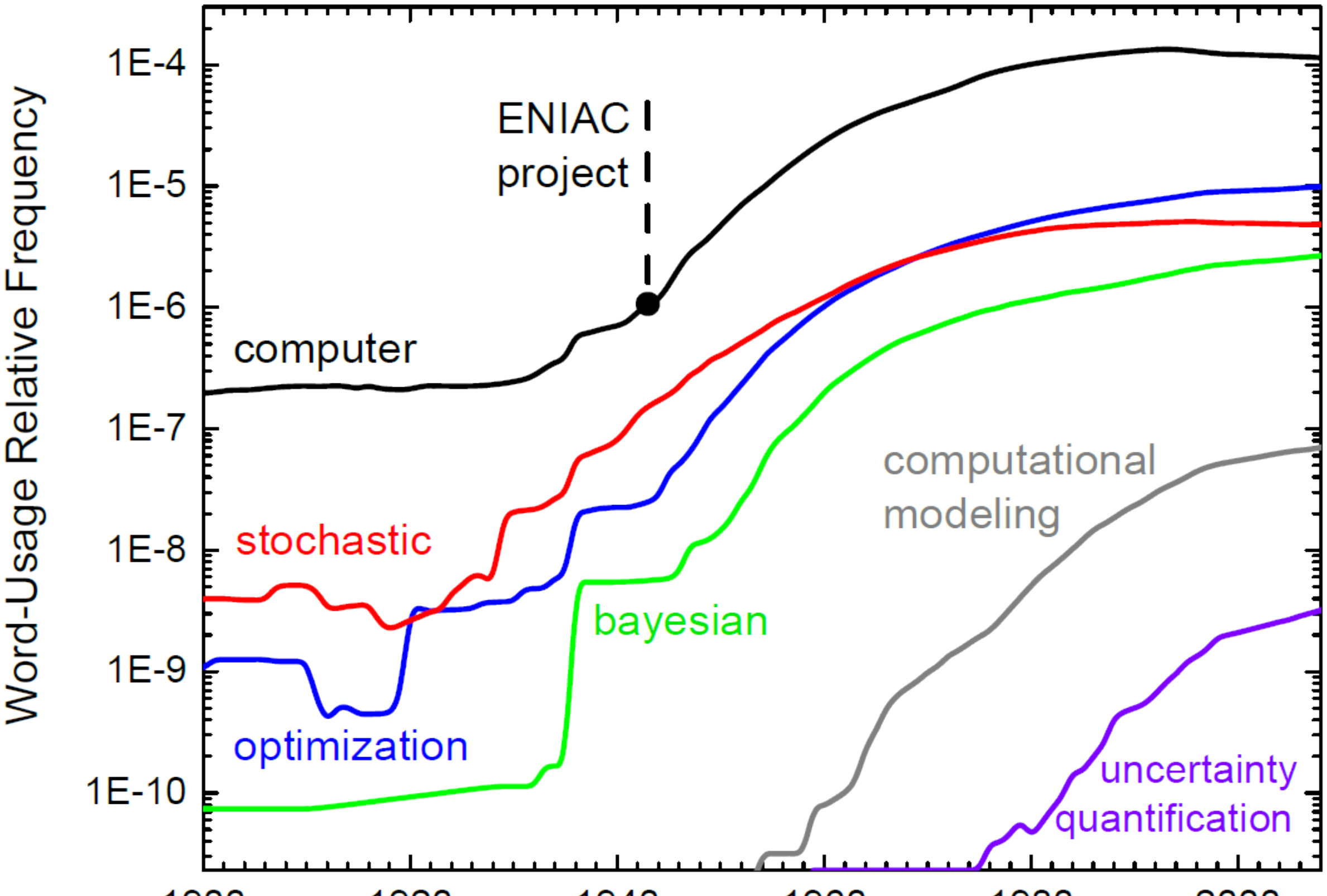
“Computer”

- ENIAC (Electronic Numerical Integrator And Computer)- one of the earliest electronic general-purpose computers made.

- Underlying structure of computer:

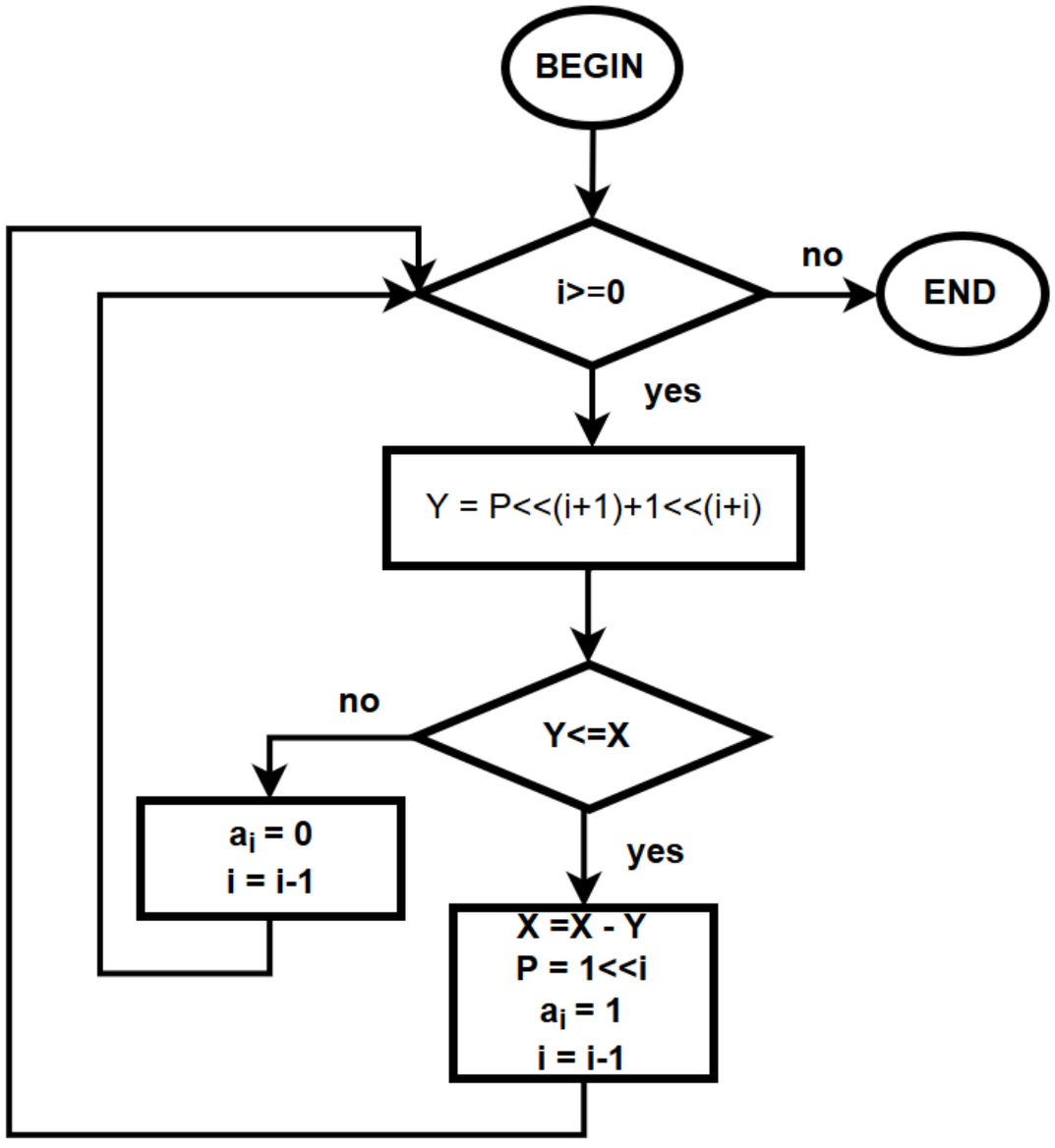


Word-usage relative frequency



Algorithms

- Human knowledge can be divided into two classes:
  - declarative. Examples:
    - J is the tenth letter of the alphabet, or,
    - Washington is the capital of the USA, or,
    - x is square-root of y only and only if $x*x=y$
- imperative (procedural).
  - a recipe on how to do something → algorithm
  - start, a flow control, and a stop.



Fixed Program Computers

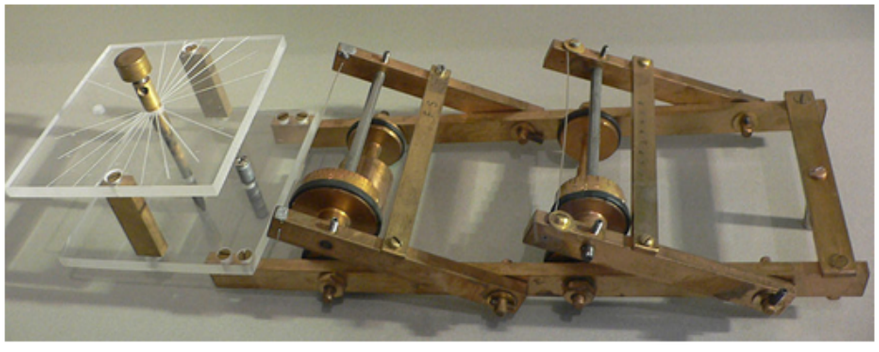
- Originally the algorithms had to be physically and mechanically implemented in computers (e.g. ENIAC)
    - the earliest computing machines were often called fixed-program computers
    - they could only perform the algorithms for which they had been physically wired and built.
    - hard to reprogram
- e.g. Desk calculator

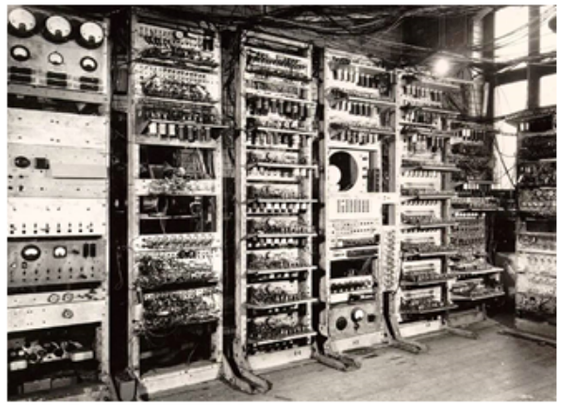
Enrico Fermi
Physicist (1901-1954)

Stanislaw Ulam
Mathematician (1909-1984)

John von Neumann
Mathematician (1903-1957)

Fermi's Analog Computer (**FERMIAC**)





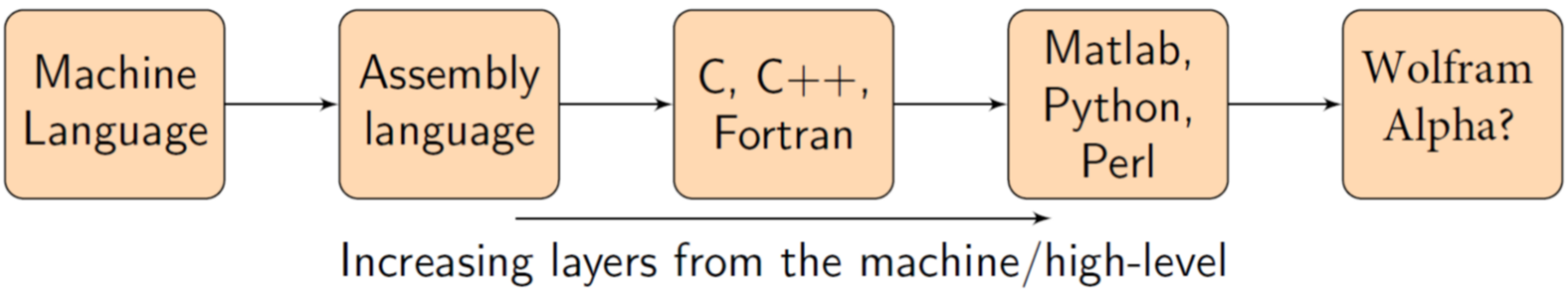
Electronic **N**umerical **I**ntegrator **A**nd **C**omputer (**ENIAC**)

General Purpose Computers

- Algorithm is also data input to the computer.
    - → Also known as stored program computers.
- Built from basic set of operations.
    - Machine code or machine language
        - a set of instructions for the Central Processing Unit (CPU) of the computer
        - comprising a long sequence of binary digital zeros and ones
        - very tedious and time-consuming, and non-portable
- → Programming languages
    - Provide higher levels of abstractions that hide complexities of machine code and interaction with the machine hardware

Generations of Programming Languages

- Increasing in level of abstraction:
    - First generation: no abstraction in interactions with computer hardware. Directly interact with computer hardware. No need for a compiler or assembler.
    - Second generation: require an assembler to interpret the code for the computer hardware. e.g. Assembly
    - Third generation: make programming almost platform-independent, e.g. Fortran, ALGOL, COBOL, BASIC, C, C#, C++, Java, Pascal
    - Fourth generation: The definition for the fourth generation and beyond is not very clear, e.g. Python, Perl, Ruby, IDL, R, S.



Mother Tongues

Tracing the roots of computer languages through the ages

Just like half of the world's spoken tongues, most of the 2,300-plus computer programming languages are either endangered or extinct. As powerhouses C/C++, Visual Basic, Cobol, Java and other modern source codes dominate our systems, hundreds of older languages are running out of life.

An ad hoc collection of engineers-electronic lexicographers, if you will-aim to save, or at least document the lingo of classic software. They're combing the globe's 9 million developers in search of coders still fluent in these nearly forgotten lingua frangas. Among the most endangered are Ada, APL, B (the predecessor of C), Lsp, Oberon, Smalltalk, and Simula.

Code-raker Grady Booch, Rational Software's chief scientist, is working with the Computer History Musuem in Silicon Valley to record and, in some cases, maintain languages by writing new compilers so our ever-changing hardware can grok the code. Why bother? "They tell us about the state of software practice, the minds of their inventors, and the technical, social, and economic forces that shaped history at the time," Booch explains. "They'll provide the raw material for software archaeologists, historians, and developers to learn what worked, what was brilliant, and what was an utter failure." Here's a peek at the strongest branches of programming's family tree. For a nearly exhaustive rundown, check out the Language List at *HTTP://www.informatik.uni-freiburg.de/Java/misc/lang_list.html*. - Michael Mendeno

Key

- 1954 **Year Introduced**
- **Active:** thousands of users
- **Protected:** taught at universities; compilers available
- **Endangered:** usage dropping off
- **Extinct:** no known active users or up-to-date compilers
- **Lineage continues**

1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001

Fortran — Once one of the most widely used languages in science and engineering.

Fortran IV — Created for the IBM 7090/94

Fortran 90 ISO/IEC

Flow-Matic — Originally B-O. One of the first English-like data-processing languages.

Cobol — The culprit behind many Y2K bugs. Also known as Common Business-Oriented Language.

Cobol 61

ANSI Cobol 74

ANSI Cobol 85

OO Cobol

IPL — Possibly the first list-processing language used in AI guru Ed Feigenbaum's PhD thesis.

Lisp — Invented by John McCarthy at MIT, Lisp has an unusual syntax made up of lots of nested parantheses. Still popular with AI researchers.

George Radin's attempt to combine the best features of Fortran, Cobol, and Algol 60.

A Lisp-like language designed by MIT researcher Seymour Papert to make programming accessible to children through graphics and simple syntax.

A Lisp offshoot that's still used in introductory programming classes at universities.

Scheme

Scheme 84

Scheme IEEE

Scheme R5RS

Haskell 1.0

Haskell 1.1

Logo

Object Logo

Clos — Common Lisp Object System.

Common Lisp

ANSI Common Lisp

PL/1

ANSI PL/1

Algorithmic Language, designed as a portable language for scientific computations. Algol 68's complexity was either ahead of its time or rightfully doomed, depending on whom you ask.

PL/M — Programming Language/Microcomputers

Self — A simple object-oriented language.

Self 4.0

The version Xerox released to the public.

Smalltalk

Smalltalk-80

NetRexx

Initially called Oak. Still one of the fastest-growing languages around, despite the standards fued between Sun and Microsoft. Somewhat like C++, Java allows for "write once, run anywhere" portability across the Net.

Algol 58 — The hoped-for Esperanto of the computing world. Designed by an international Zurich-based committee as a universal language, it was one of the first attempts at making software more portable. Initially called International Algebraic Language.

Algol 68

The first language to show off the power of object-oriented code. Developed by Software Concepts Group, Xerox PARC, led by Alan Kay.

Rex 1.00

Rex 2.00

Object Rexx

Restructured Extended Executor

An object-oriented language.

A simple language (A = argument, B = basic varue, C = ?).

Interactive object-oriented language.

Simula — Popular in Europe during the '70s, Simula introduced the now-standard concept of object-oriented, rather than procedural, programming. Desgined by Kristen Nygaard and Ole-Johan Dahl.

Simula 67 — All-purpose version of Simula

Java

Java 2(v1.3)

Prolog — Programming en Logique for natural-language processing. Popular for AI programs.

Clu — Called Cluster, it's an object-oriented academic language created to teach rigid engineering skills.

Marco and text-processing language used by Unix admins to manipulate large config files programmatically, rather than editing them by hand.

Objective-C

Eiffel

Eiffel 3

Microsoft's answer to Java, it is a key component of the Microsoft.Net platform for Web services.

ABC

Sather 0.1

Ruby — An object oriented language.

New AWK, a pattern scanning and processing language.

Python

Python 1.6

AWK

NAWK

A popular language among Web site builders, it includes features missing from Perl.

ECMA Script

Possibly the most commen language today. Adds object-oriented features to C.

Tool Command Language, "tickle." The duct tape of programming (a scripting language for patching together different languages).

Cmm — C--, a scripting language.

JavaScript

JScript

Microsoft's version of JavaScript. Can't they share anything? :-)

C#

One of the most widely deployed languages today. Windows and Unix OSes are written mostly in C and its descendants.

C with Classes

C++

Internet C++

Concurrent C

Tcl

Tcl/Tk

Tcl/Tk 8.1

BCPL — Basic Combined Programming Language

B — Used primarily for non-numeric programming.

C

ANSI C (C89)

ISO C (C95)

Updates C++ for the Net with a Java-like virtual machine, so code can run on any kind of computer.

ADA

ADA 83

ADA 95

The US Department of Defense's effort to craft a standard object-oriented language for its work. Named after Ada Lovelace, arguably the world's first computer programmer, and created by Jean Ichbiah's team of Honeywell.

Modula 3

Found in millions of Web pages. Originally dubbed LiveScript, it was renamed by Netscape marketers who licensed the name to ride Java's buzz. It has little in common with that language.

A teaching language named for French mathematician Blaise Pascal. Designed by legendary Swiss programmer Niklaus Wirth for simplicity, in reaction to the complexity of Algol 68.

Modula — Modular Language

Modula 2 ISO

Object-oriented Pascal, designed for simplicity.

Object Pascal

Oberon

Another creation of Niklaus Wirth. Later modified by Robert Griesemer for numerical apps on supercomputers.

Delphi

Pascal

Csh

The swiss Army Knife of programming (a k a Practical Extraction and Report Language), used for patching together different languages. Spawned a quasi-literary culture that writes Perl haiku

Perl 1.000

Perl 4.000

sh

Ksh

Frame-based language.

C-Shell, a scripting language and command-shell interpreter. Written by C programmers to make Unix command lines more like C's.

Standard ML, a general-purpose language.

The kitchen sink of command-line programming features for Unix, based on the original Bourne shell (a k a Bourne Again Shell).

Bash

String-processing language for text and formula manipulation, common in text processers. Invented by David Farber, Ralph Griswold, and Ivan Polonsky at Bell Labs.

Programmer's easy street - with this descendant of Snobol, there's no need to know the underlying OS.

A general-purpose language known for its beauty and grace. Designed by Ralph Griswold to be successor to Snobol4.

Snobol

Snobol4

Icon

ML — Meta Language

SML

SML 90

SML 97

King of the one-liners, also referred to as A Programming Language. An algebra-like language with a special set of characters for writing complex programs as short formulas. Designed by Kenneth Iverson at Harvard.

Word-based language first used to guide the National Radio Astronomy Observatory telescope at Kitt Peak in Arizone.

Page-description language for printers and graphics sistems; 75 percent of all commercial documents are produced on PostScript printers.

Caml

Caml 2-6.1

Objective Caml

O Caml 2

O Caml 3.00

Forth

PostScript

PostScript Level 2

J

Categorical Abstract Machine Language

Although mocked by "real" programmers for its limited abilities, Basic has outlived many more advanced languages, as well as the RadioShack TRS-80 computers that made it a household word. Stands for Bigenner's All-Purpose Symbolic Instruction Code.

APL

APL 96

Microsoft Basic

Basic

MS Basic 2.0

Visual Basic

The Rodney Dangerfield of programming languages. Popular for building Web sites with Microsoft Visual Studio tools.

Survival of the Fittest

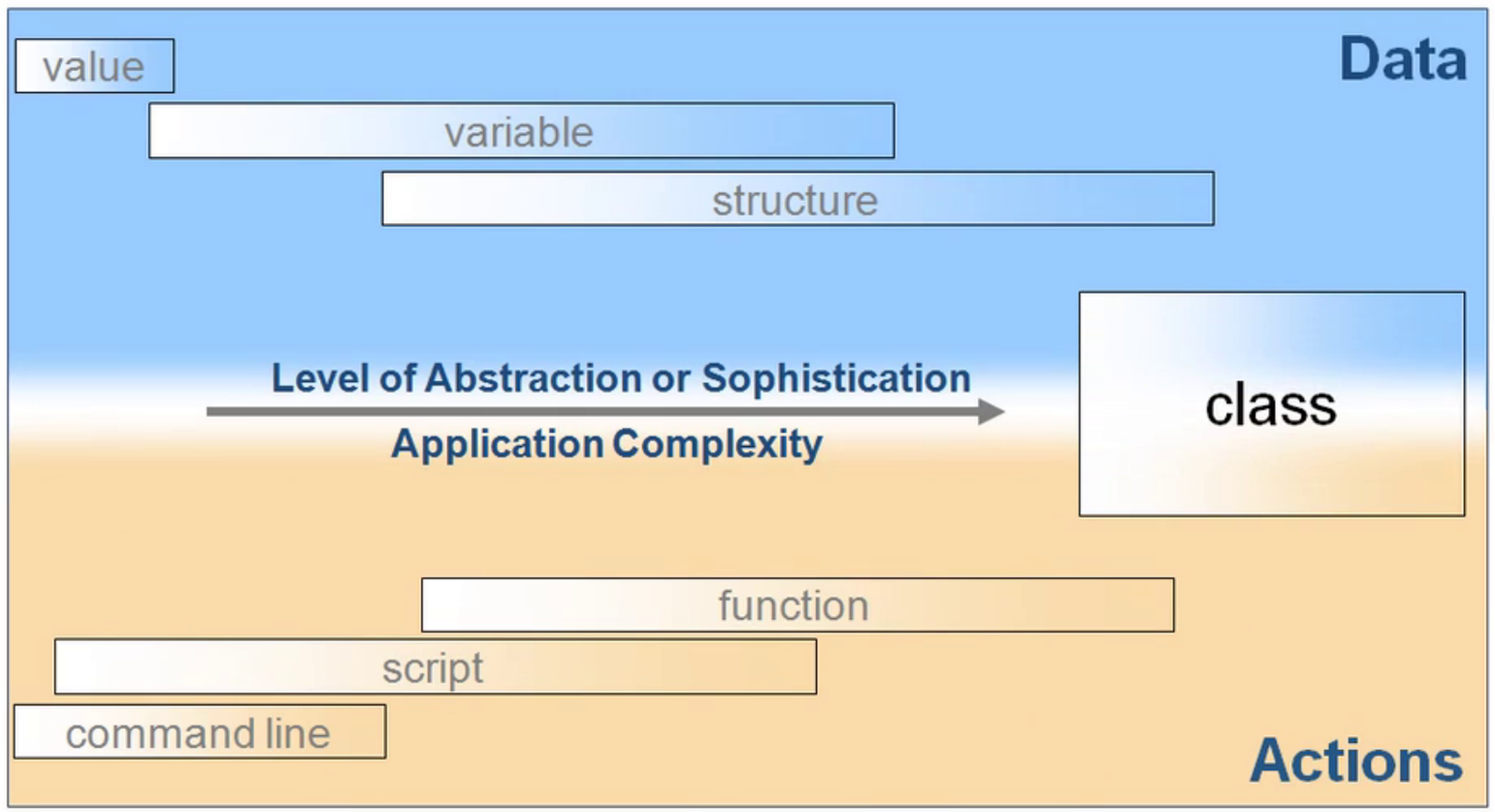
Reasons a language endures, with examples of some classic tongues

- **Appeals to a wide audience** C (bolstered by the popularity of Unix)
- **Gets a job done** Cobol (designed for business-report writing)
- **Delivers new functionality** Java (runs on any hardware platform)
- **Fills a niche** Mathematica (speeds up complex computations)
- **Offers a modicum of elegance** Icon (has friendly, line-oriented syntax)
- **Has a powerful user base or backer** C# (developed by Microsoft for .Net)
- **Has a charismatic leader** Perl (programmer-author Larry Wall)

Sources: Paul Boutin; Brent Hailpern, associate director of computer science at IBM Research; The Retrocomputing Museum; Todd Proebsting, senior researcher at Microsoft; Gio Wiederhold, computer scientist, Stanford University

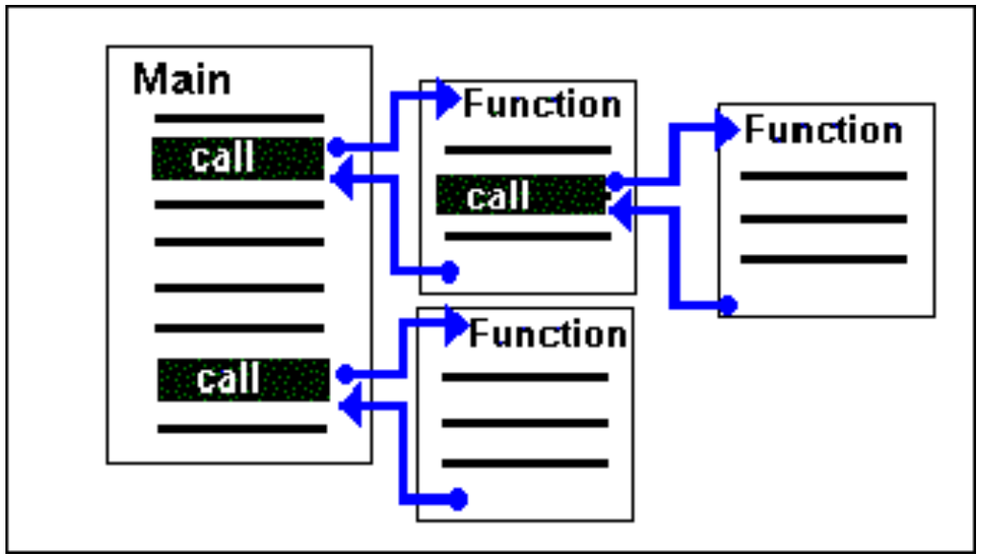
Programming language paradigms

- structured programming,
- imperative programming (allows side effects),
- functional programming (disallows side effects),
- procedural programming (groups code into functions and subroutines),
- object-oriented programming (OOP) (groups code together with the data on which the code works).



Imperative Programming

- Imperative programs components and rules:
  - contain a main procedure that determines the control flow for the program
  - contain a set of functions that are called to perform certain tasks during program execution
  - the main and sub-main procedures have a hierarchical structure
  - the source code for each procedure is compiled
  - all compiled procedures and the main procedure are linked together with other source codes to produce a complete executable program.



Object Oriented Programming

- Combine code and data into objects.
  - Class - the type of object
    - Attributes
    - Methods
  - Instance