

Programming – Lecture 1

Feedback from a recent conference: As a virtual participant, I followed the overall trend of disabling my webcam outside of my own talk. This makes it sooooo tempting to do other work, which of course is always detrimental to a conference. It may be a good move to motivate the virtual participants to keep their cam active in order to maintain a critical mass of visible faces.

***Sooo ... At lecture start time (or before) –
please show us your beautiful face, thanks!***

Your Privacy

- In the end, it's of course up to you whether you switch on your camera or not.
- However, seeing your faces allows some “class atmosphere” for you, and it would help me to judge how well things come across.
- In any case, it is forbidden by law to make photos/recordings of participants.
- Also, it's up to you whether you participate with your actual name or with some alias.
- However, seeing your real name may help you to find buddies, and it would allow me to give you credit for contributions to the class.

Interaction Protocol

- Per default, please have your microphone muted.
- If you want to ask something/respond to a question, raise your hand – either on video, or by using “raise hand” feature in zoom.
- To take your comment, I will not mention your name (to preserve anonymity), but will prompt you to unmute your microphone (a msg box will pop up on your screen)
- After you made your comment, lower your hand (if you used the zoom feature), and mute your mike again.
- If it's urgent or if I appear to overlook you for a longer time, you may also just switch on your mike and speak up.

Programming – Lecture 1

Welcome

Administrivia

Introduction (Chapter 1)

- Brief history of computing
- CS, Algorithms, SW Engineering, Errors
- Compilation vs. Interpretation

We

Lecturer:

Prof. Reinhard von Hanxleden
R1117, rvh@informatik.uni-kiel.de

Staff (*Mitarbeiter*):

Steven Smyth (lead)
ssm@

Niklas Rentz
nre@

Alexander Schulz-Rosengarten
als@

Student Staff

(*Hilfskräfte*):

Liam David Boddin

Sascha Ecks

Christopher Konkel

Tim Krawczyk

Patrik Michalski

Kajus Park

Alexander Pfanne

Hendrik Sauer

Björn Schumacher

You?

Please visit

<http://pingo.upb.de/643250>

Literature

- "The Art and Science of Java,"
by Eric Roberts – our "*Skript*"
- "Java ist auch eine Insel,"
by Christian Ullenboom
- See also wiki \leadsto Begleitmaterialien

Code of Conduct

I:

- Start and finish on time
- Try to deliver high-quality lectures
- Listen to your concerns
- Make sure you have a fair chance of passing this class
- Prepare you for what's ahead

You:

- Are **punctual**
- Do not disturb others (**laptops/tablets only in last row**)

Wait – we're in virtual space anyway, so scratch that ...

- Work on problems yourself (**no plagiarism/code sharing**)
- Familiarize yourself with infprogoo.de and the linked wiki, including requirements for "*Endklausurzulassung*"

Programming – Lecture 1

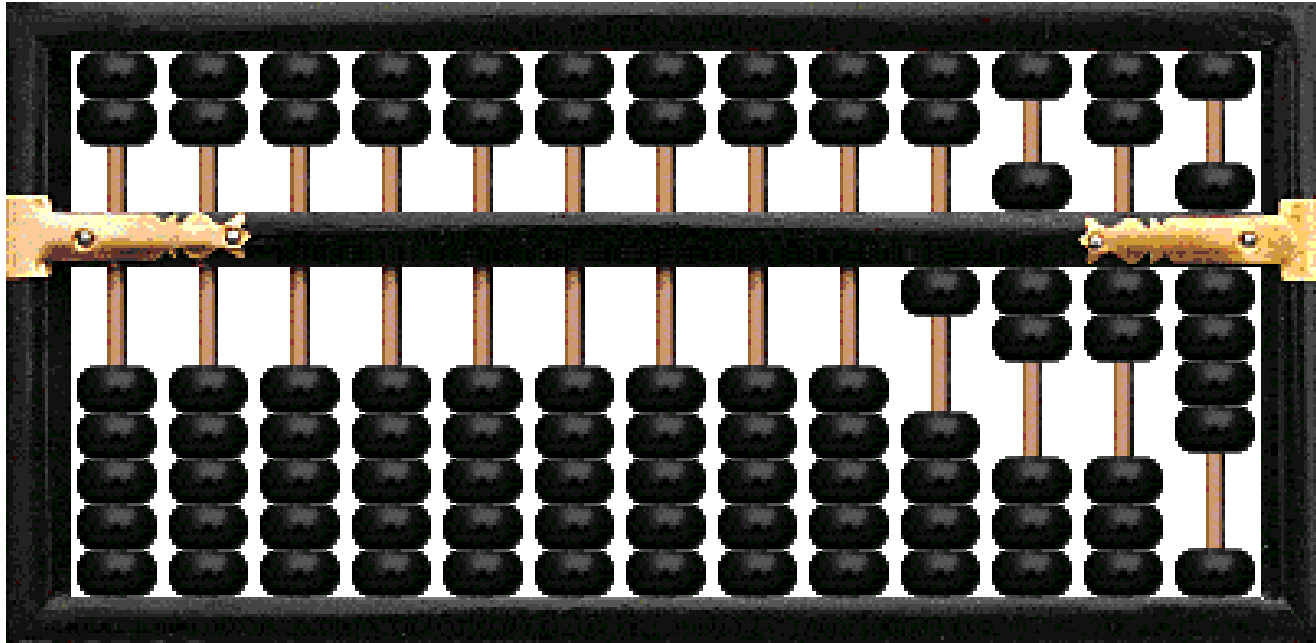
Welcome

Administrivia

Introduction (Chapter 1)

- Brief history of computing
- CS, Algorithms, SW Engineering, Errors
- Compilation vs. Interpretation

A Brief History of Computing





Charles Babbage
(1791 – 1871)



Augusta Ada Byron,
Lady Lovelace
(1815 – 1852)



CC BY-SA 3.0

Konrad Zuse
(1910 – 1995)



CC BY-SA 3.0

Z1

Computer Science

The science of problem solving in which the solutions happen to involve a computer

Solving a problem by computer:

1. Algorithmic design
2. Coding

Algorithm

Ninth-century Persian mathematician Abu Ja'far Mohammed ibn Mûsâ al-Khowârizmî

Properties:

1. Clearly and unambiguously defined
2. Effective
3. Finite

Software Engineering

Discipline of writing programs so that they can be understood and maintained by others

Programming Errors

Distinguish

- Syntax errors
- Bugs

*„The first step [in all of my inventions] is an intuition, and comes with a burst, then difficulties arise – this thing gives out and [it is] then that ‘**Bugs**’ – as such little faults and difficulties are called – show themselves [...].“*

Thomas Edison, 18 Nov 1878

9/9

0800 Antan started
 1000 " stopped - antan ✓

			{ 1.2700	9.037 847 025
				9.037 846 995 connect
	13" oc (032)	MP - MC	1.982147000	
	(033)	PRO 2	2.130476415	4.615925059(-2)
		connect	2.130676415	

Relays 6-2 in 033 failed special speed test
 in relay .. 11.00 test.

Relay
 2145
 Relay 337

1100 Started Cosine Tapc (Sine check)
 1525 Started Mult + Adder Test.

1545



Relay #70 Panel F
 (moth) in relay.

First actual case of bug being found.

~~1630~~ 1630 antan started.
 1700 closed down.

[U.S. Naval Historical Center Online Library]

Page from the Harvard Mark II
 electromechanical computer's log, 1947

Why Java in InfProgOO?

- First of all: why ask that question? This class is about *programming principles*, not about a particular *programming language*.
- Having said that: to really learn the principles, one should write real programs, for which a concrete language X must be chosen.
- Choice of X should be guided by a few questions
...

Questions to Ask Concerning X

1. Does X illustrate imperative/object-oriented programming?
2. Is it reasonably easy to learn X, are there good resources available for learning X?
3. Does the teaching staff have good expertise on X?
4. Do later classes and software projects at CAU also use X?
5. Does knowledge of X help me after graduation?

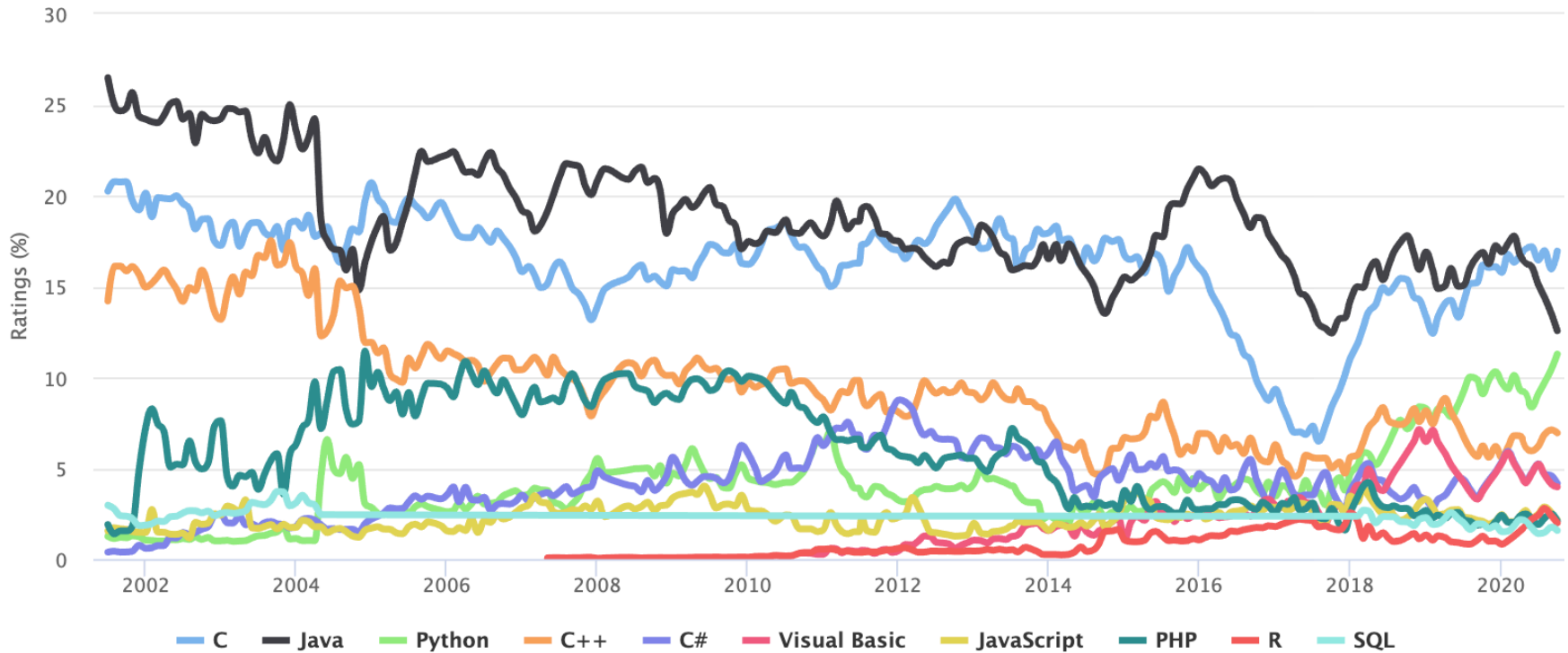
Clear “yes” to all of these for X = Java.

Programming Language	2020	2015	2010	2005	2000	1995	1990	1985
C	1	2	2	1	1	2	1	1
Java	2	1	1	2	3	29	-	-
Python	3	6	6	6	21	15	-	-
C++	4	3	3	3	2	1	2	9
C#	5	4	5	7	9	-	-	-
JavaScript	6	8	8	10	7	-	-	-
PHP	7	7	4	5	19	-	-	-
SQL	8	-	-	-	-	-	-	-
Swift	9	16	-	-	-	-	-	-
R	10	13	49	-	-	-	-	-
Lisp	29	25	15	13	8	5	6	2
Fortran	31	24	24	15	15	17	3	5
Ada	33	27	22	17	17	4	7	3
Pascal	242	15	14	16	16	3	10	6

www.tiobe.com

TIOBE Programming Community Index

Source: www.tiobe.com



See also <http://www.tiobe.com/tiobe-index/programming-languages-definition/>

Java – the undisputed winner

Java still continues to top the most popular programming languages charts as it did a year ago. According to TIOBE's data, Java has secured the first and second positions more than any other languages for about a couple of decades. A large number of renowned companies use Java to develop software and applications so if you happen to know Java, you definitely won't have to struggle to find a job. The major reasons behind the popularity of Java are its portability, scalability and a large community of users.

<https://fossbytes.com/most-popular-programming-languages/>

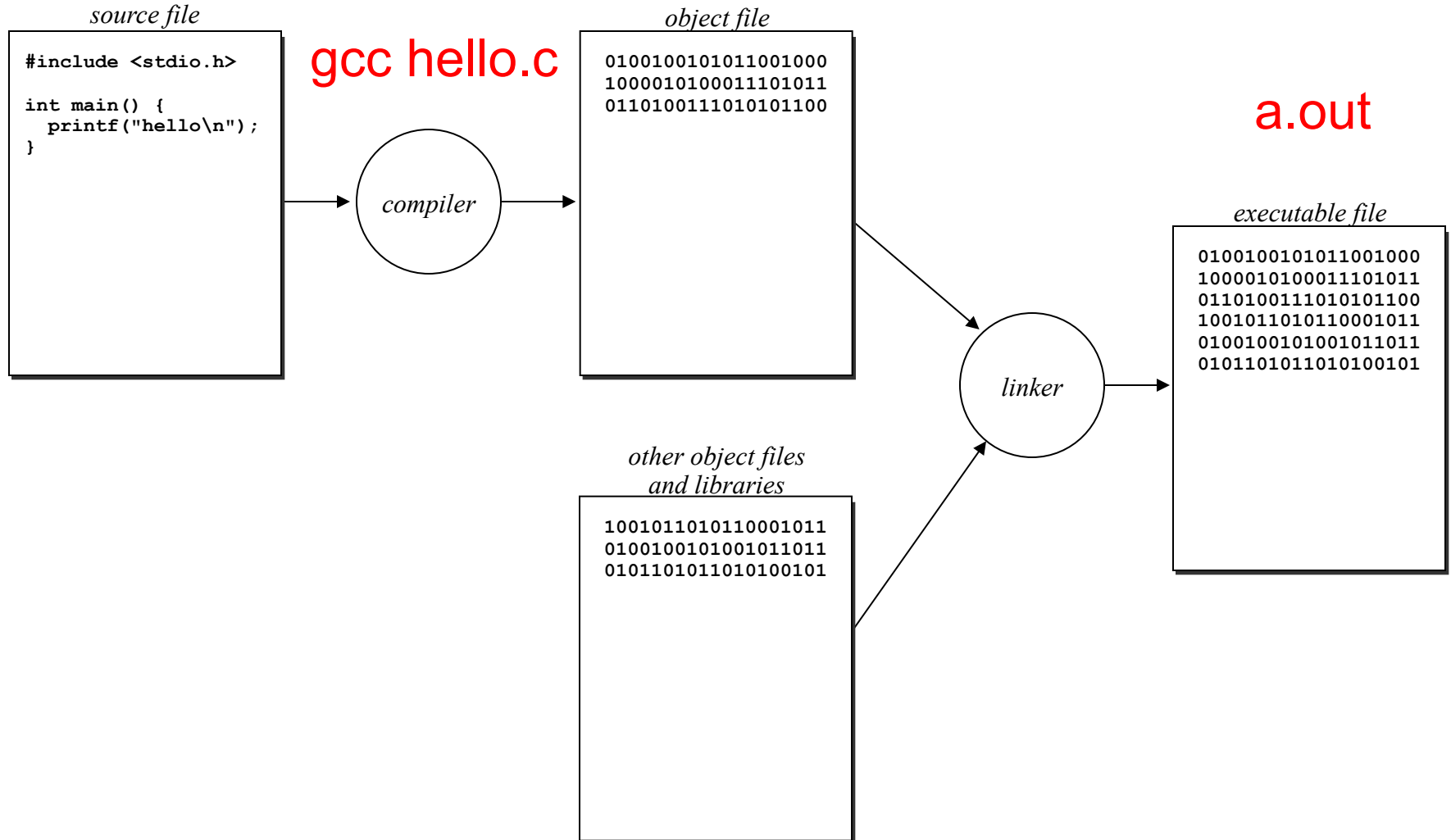
June 2018

Java – Design Goals

- Simple, object oriented, familiar
- Robust, secure
- Architecture neutral, portable
- High performance
- Interpreted, threaded, dynamic

White Paper: The Java Language Environment
James Gosling and Henry McGilton, May 1996

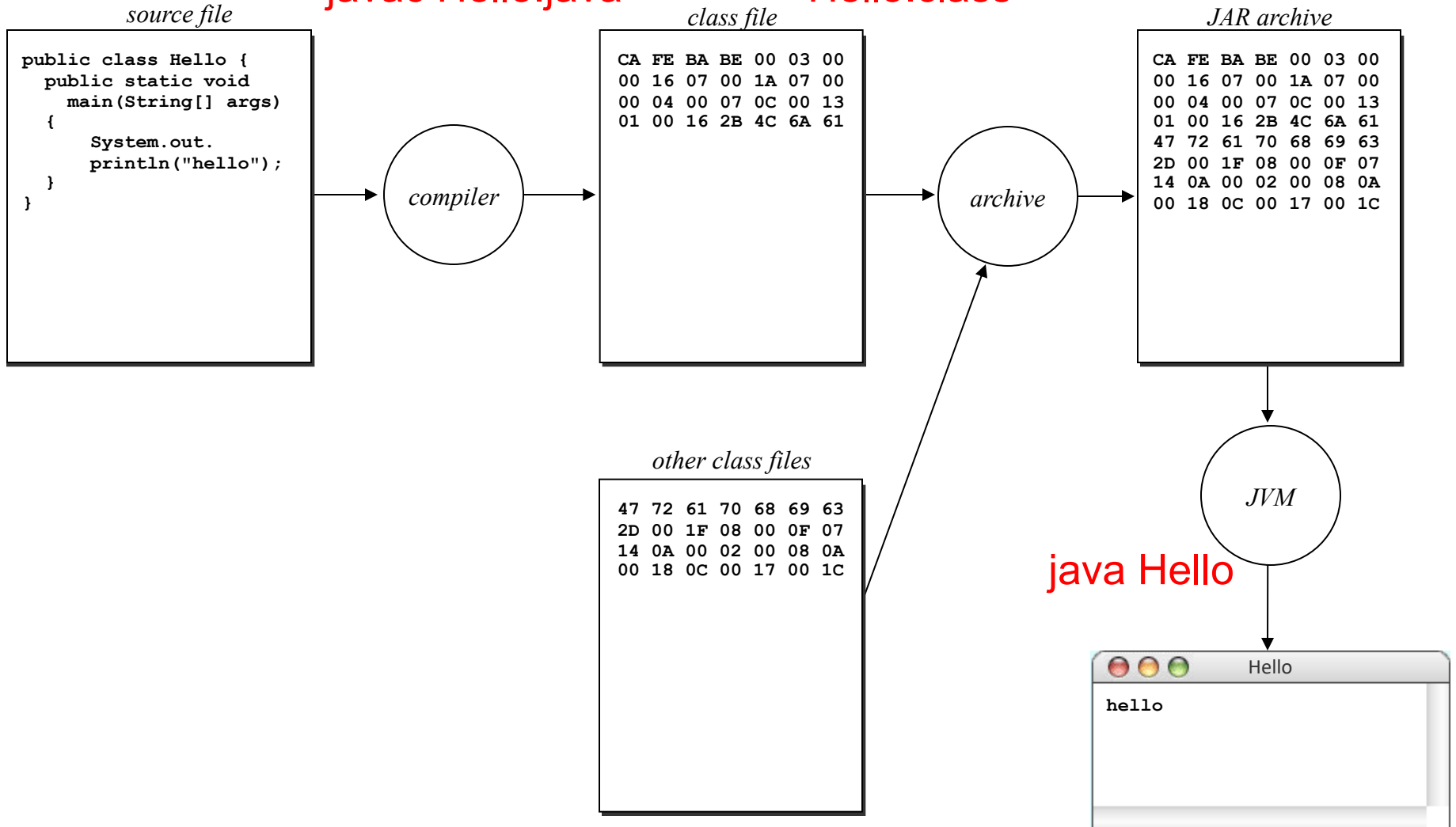
Classic Compilation Process



The Java Interpreter

javac Hello.java

Hello.class



Summary

- Long history of computing, recent acceleration
- Defined “CS”, “Algorithm”, “SW Engineering”
- Syntax errors vs. Bugs
- Discussed Java design goals
- Compilation vs. Interpretation