Inf2C - Computer Systems
Lecture 1
Course overview & the big picture

Boris Grot

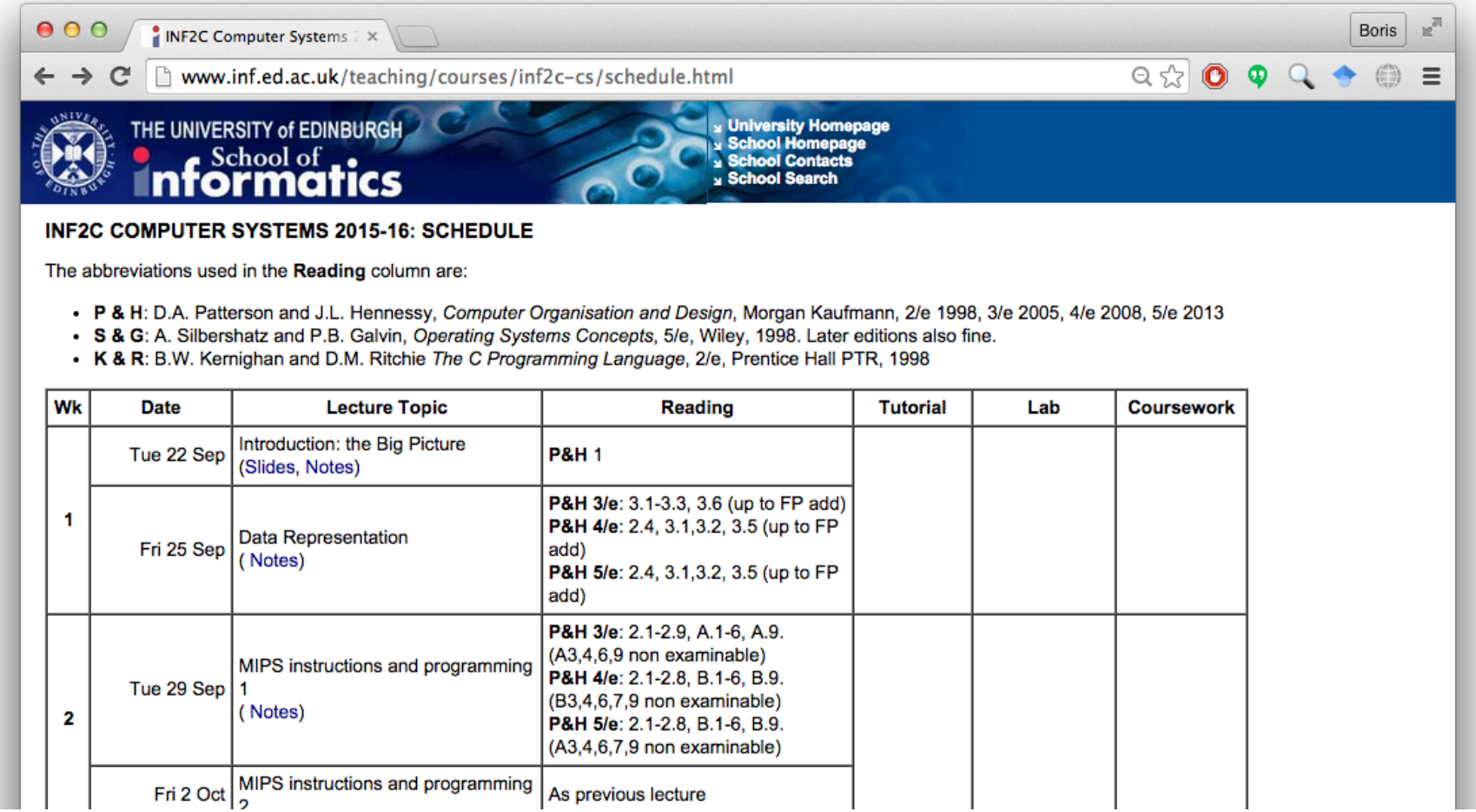
School of Informatics

University of Edinburgh

Practicalities

- Lectures:
  - Tue & Fri, David Hume Tower, Lec. Hall C @ 15:10– 16:00
- Tutorials: weeks 3, 5, 7, 9
- Drop-in labs: demonstrators available to help
- Online discussion forum: TBD
  - Primary means to Q&A outside of class.
- Notes are provided, but must read the book too
- All material are/will be on the course web-page: http://www.inf.ed.ac.uk/teaching/courses/inf2c-cs
  - Previous year's materials also online

Lecture schedule, slides, notes



INF2C Computer Systems

Boris

www.inf.ed.ac.uk/teaching/courses/inf2c-cs/schedule.html

THE UNIVERSITY of EDINBURGH School of informatics

University Homepage
School Homepage
School Contacts
School Search

INF2C COMPUTER SYSTEMS 2015-16: SCHEDULE

The abbreviations used in the **Reading** column are:

- **P & H**: D.A. Patterson and J.L. Hennessy, *Computer Organisation and Design*, Morgan Kaufmann, 2/e 1998, 3/e 2005, 4/e 2008, 5/e 2013
- **S & G**: A. Silbershatz and P.B. Galvin, *Operating Systems Concepts*, 5/e, Wiley, 1998. Later editions also fine.
- **K & R**: B.W. Kernighan and D.M. Ritchie *The C Programming Language*, 2/e, Prentice Hall PTR, 1998

Wk	Date	Lecture Topic	Reading	Tutorial	Lab	Coursework
1	Tue 22 Sep	Introduction: the Big Picture (Slides, Notes)	**P&H** 1			
	Fri 25 Sep	Data Representation (Notes)	**P&H 3/e**: 3.1-3.3, 3.6 (up to FP add) **P&H 4/e**: 2.4, 3.1,3.2, 3.5 (up to FP add) **P&H 5/e**: 2.4, 3.1,3.2, 3.5 (up to FP add)			
2	Tue 29 Sep	MIPS instructions and programming 1 (Notes)	**P&H 3/e**: 2.1-2.9, A.1-6, A.9. (A3,4,6,9 non examinable) **P&H 4/e**: 2.1-2.8, B.1-6, B.9. (B3,4,6,7,9 non examinable) **P&H 5/e**: 2.1-2.8, B.1-6, B.9. (A3,4,6,7,9 non examinable)			
	Fri 2 Oct	MIPS instructions and programming 2	As previous lecture			

Schedule will drift. It's OK.

Books

- **Required:** *Patterson & Hennessy: Computer Organization and Design, Morgan Kaufmann*
  - 5th or 4th ed recommended
  - Library has 2nd and 3rd ed (both OK, but try to get newer ed)
- *Silberschatz, Galvin, Gagne: Operating Systems Concepts, Wiley 9th ed*
  - Library has 5th and 7th ed ebook (both OK)
  - Only a few sections needed for this course
- *Kernighan and Ritchie. The C Programming Language, Prentice Hall 2nd ed*
  - Generally useful, but not mandatory for this course

Exam and Coursework

- Exam - 60%
  - In December; exact date not available yet.
  - Must achieve at least 35/100 to pass the course
- Coursework – 40%
  - Must achieve at least 25/100 to pass the course

1. MIPS assembly programming
   - Out: Tue 13 Oct (week 4)
   - Due: Tue 27 Oct (week 6) @ 4pm
2. TBD
   - Out: Tue 10 Nov (week 9)
   - Due: Tue 24 Nov (week 11) @ 4pm

Late coursework

- School-wide consistent policy:
  Normally, you will not be allowed to submit coursework late
- If you have a **good reason** to submit late, contact the ITO via their Support Form.
  - The ITO will log the report and pass it on to the UG2 Course/Year Organiser (Dr. Sharon Goldwater)
  - Only in exceptional circumstances (*e.g.*, illness that stopped you getting to email), would an extension be granted after a deadline has passed
- See the online Undergraduate Year 2 Handbook for details

Good reason

Something that, in the judgement of the member of staff responsible, would prevent a competent, well-organised, conscientious student from being able to submit on time.

Examples:

- Significant illness
- Serious personal problems

Non-examples:

- Difficult cluster of deadlines
- Last-minute computer problems, dog ate your homework, …

So what is this course about?

Syllabus Overview

- Hardware:
  - Data representation and operations
  - Design of (very) simple circuits
  - Processor organisation
  - Exceptions and interrupts
  - The memory subsystem
  - Input/Output (I/O)
- Software:
  - Low-level (assembly) programming
  - Operating systems basics
  - Introduction to C programming (2 lectures)

Words of wisdom & caution

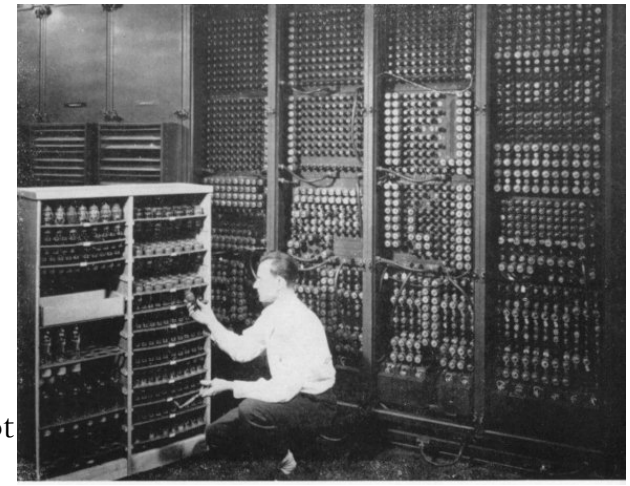
- This class covers a lot of material.
- Keeping up will require effort on your part.
  - This ain't no INF1!
- Attend all lectures, tutorials and labs.
- Seek help early.
- ASK QUESTIONS!

Reward: you will learn a lot!

Let's dig in…

Evolution of computers

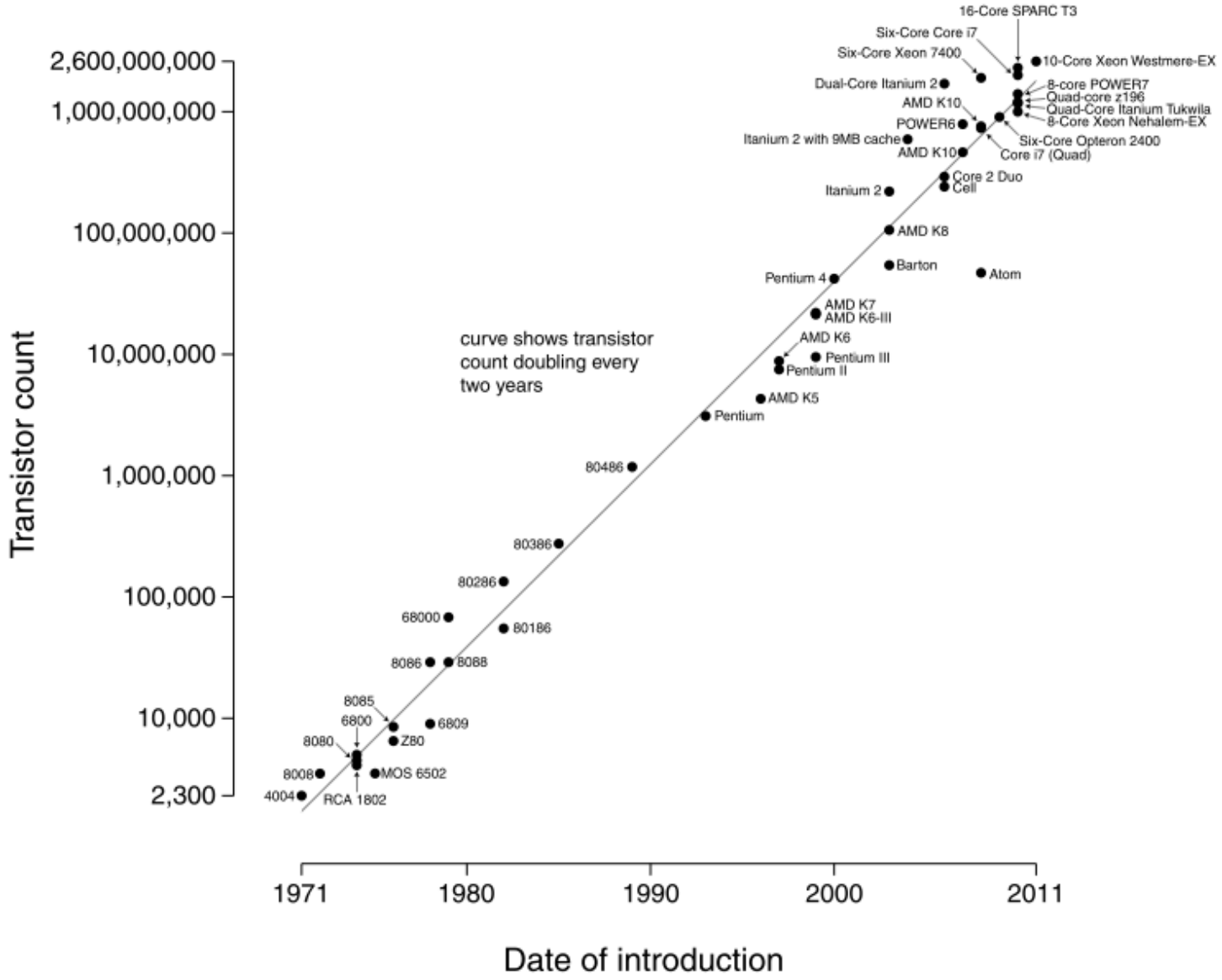
- Early computers had their programs set up by plugging cables and setting switches
- John von Neumann first proposed to store the program in the computer's memory
- All computers since then (~1945) are stored-program machines



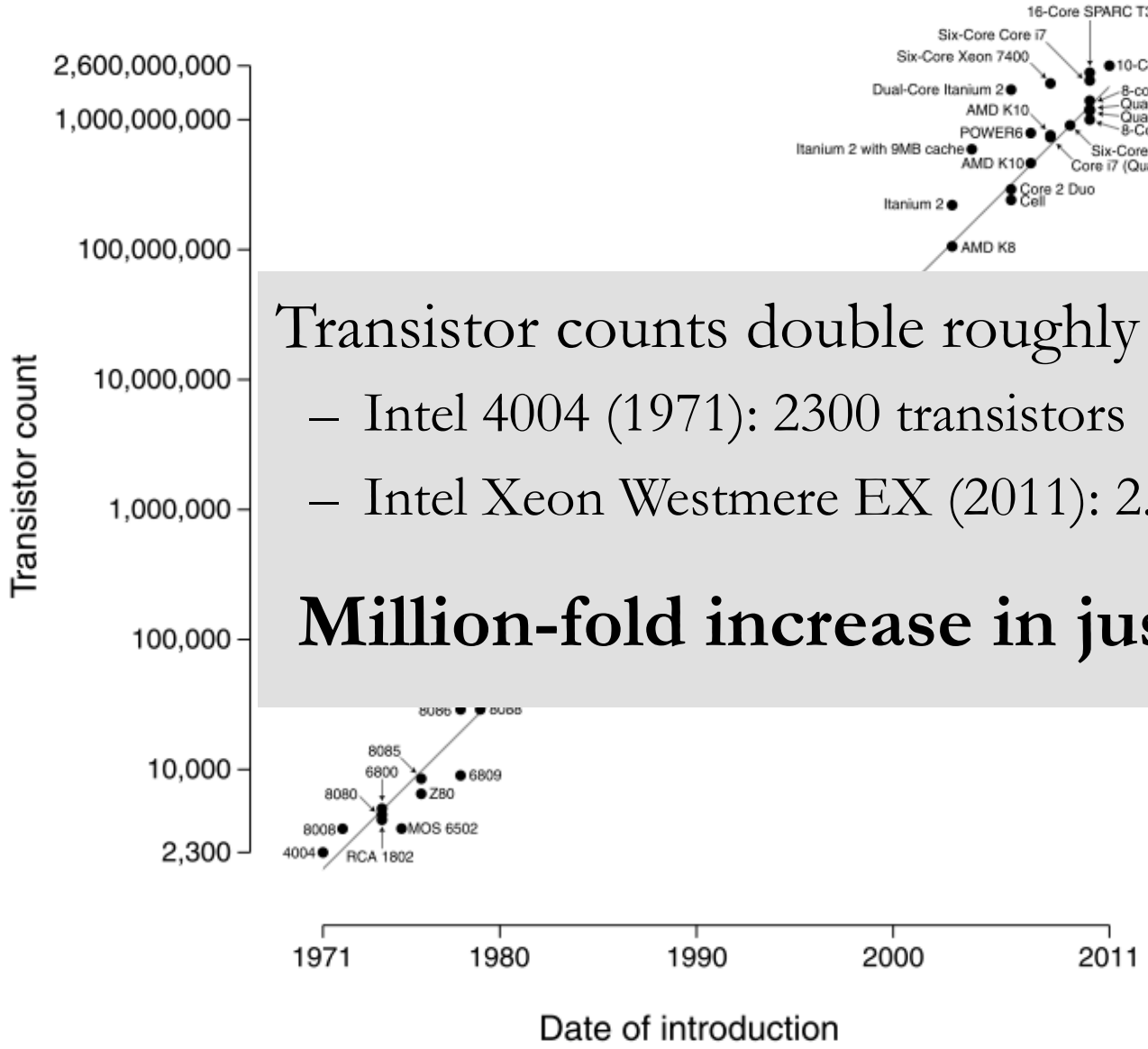
Evolution of computers

- What has changed is the number of transistors (electronic switches) and their speed
- Implementation technology progressed from vacuum tubes to discrete bipolar transistors to (eventually) Integrated Circuits (a.k.a. chips) made with complementary metal-oxide semiconductor (CMOS) technology.
- At the same time, the cost per transistor has been dropping

Moore's law


Transistor count
2,600,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
2,300
1971
1980
1990
2000
2011
Date of introduction
curve shows transistor count doubling every two years
16-Core SPARC T3
Six-Core Core i7
Six-Core Xeon 7400
10-Core Xeon Westmere-EX
Dual-Core Itanium 2
8-core POWER7
Quad-core z196
AMD K10
Quad-Core Itanium Tukwila
POWER6
8-Core Xeon Nehalem-EX
Itanium 2 with 9MB cache
Six-Core Opteron 2400
AMD K10
Core i7 (Quad)
Core 2 Duo
Cell
Itanium 2
AMD K8
Barton
Atom
Pentium 4
AMD K7
AMD K6-III
AMD K6
Pentium III
Pentium II
AMD K5
Pentium
80486
80386
80286
68000
80186
8086
8088
8085
6800
6809
8080
Z80
8008
MOS 6502
4004
RCA 1802

Moore's law



Transistor counts double roughly every 2 years

- Intel 4004 (1971): 2300 transistors
- Intel Xeon Westmere EX (2011): 2.3 bil. transistors

Million-fold increase in just 40 years!

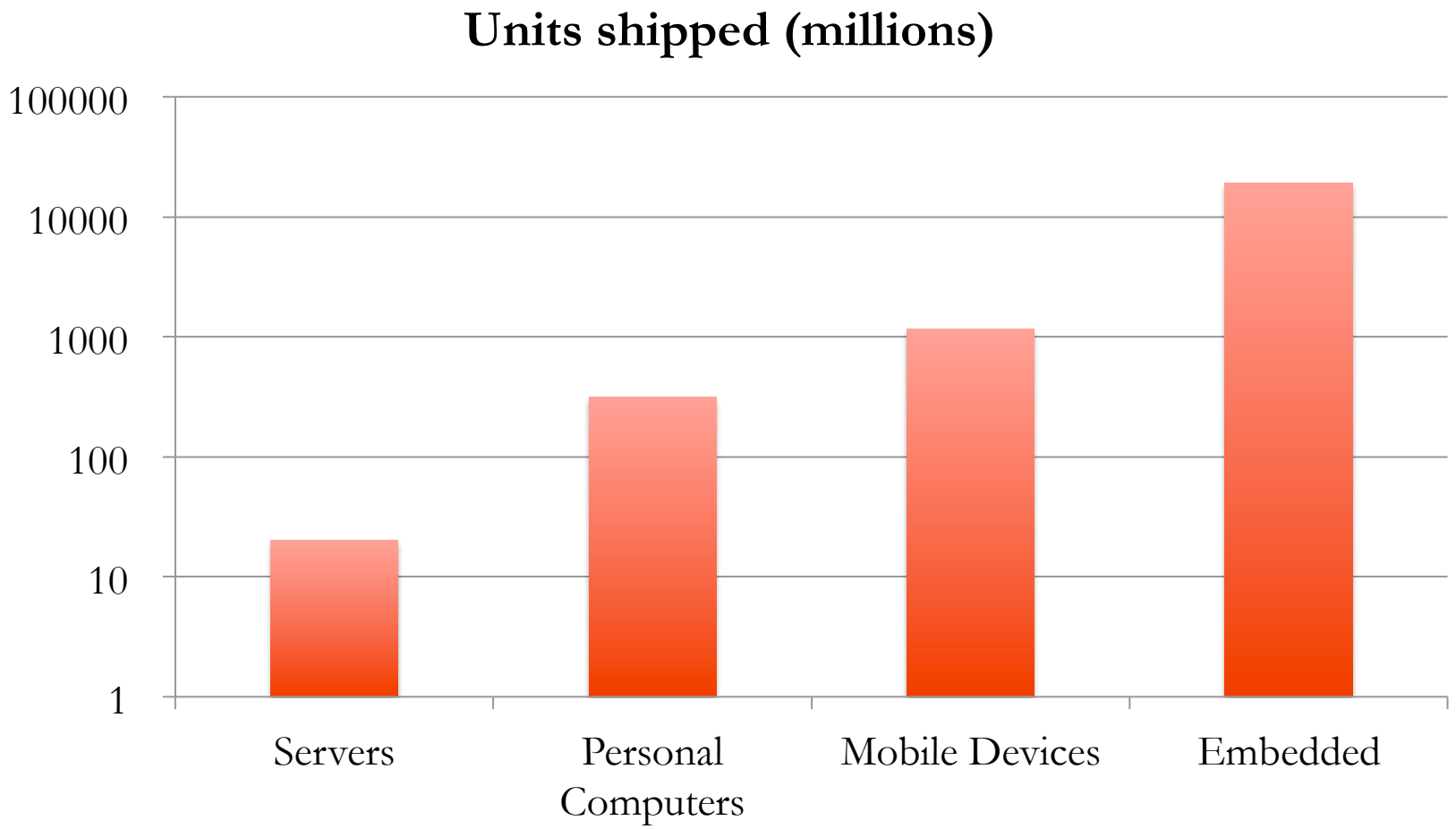
Types of computer systems

- Servers
  - Used for either few large tasks (e.g., engineering apps), or many small tasks (e.g., web server, Google)
  - Fast processors, lots of memory
  - Multi-user, multi-program
- Personal computers
  - Laptops, desktops
  - Balance cost, processing power
  - Few users, multi-program

Types of computer systems (con'd)

- Mobile devices
  - Smart phones, tablets
  - Highly integrated (multiple processors, GPU, GPS, media accelerators, etc), low-power
  - Single-user, multi-program
- Embedded:
  - Task specific: sensing, control, media playback, etc.
  - Low-cost, low-power
  - Single program

Which computer system category is the largest?

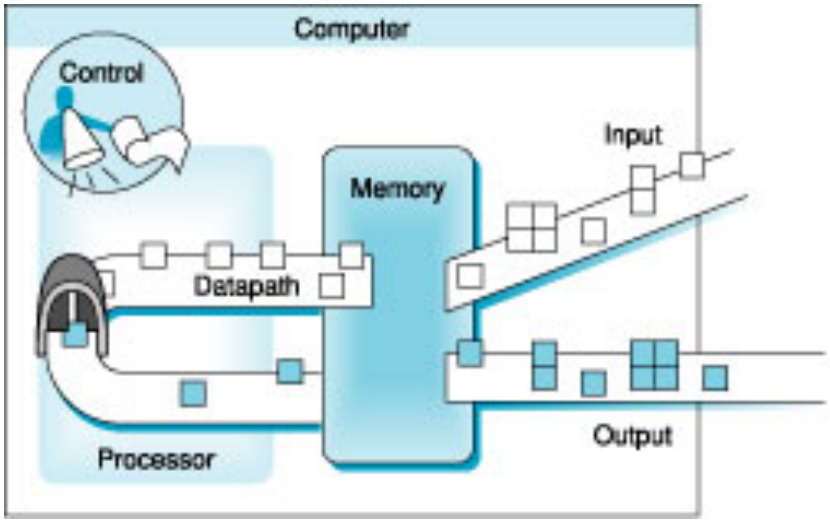


Computer components

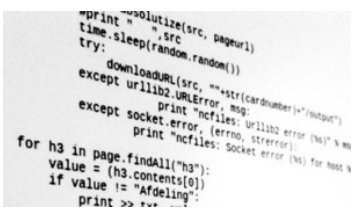
- Data path
  - Performs actual operations on data
- Control path
  - Fetches instructions from program in memory
  - Controls the flow of data through the data path

} Processor

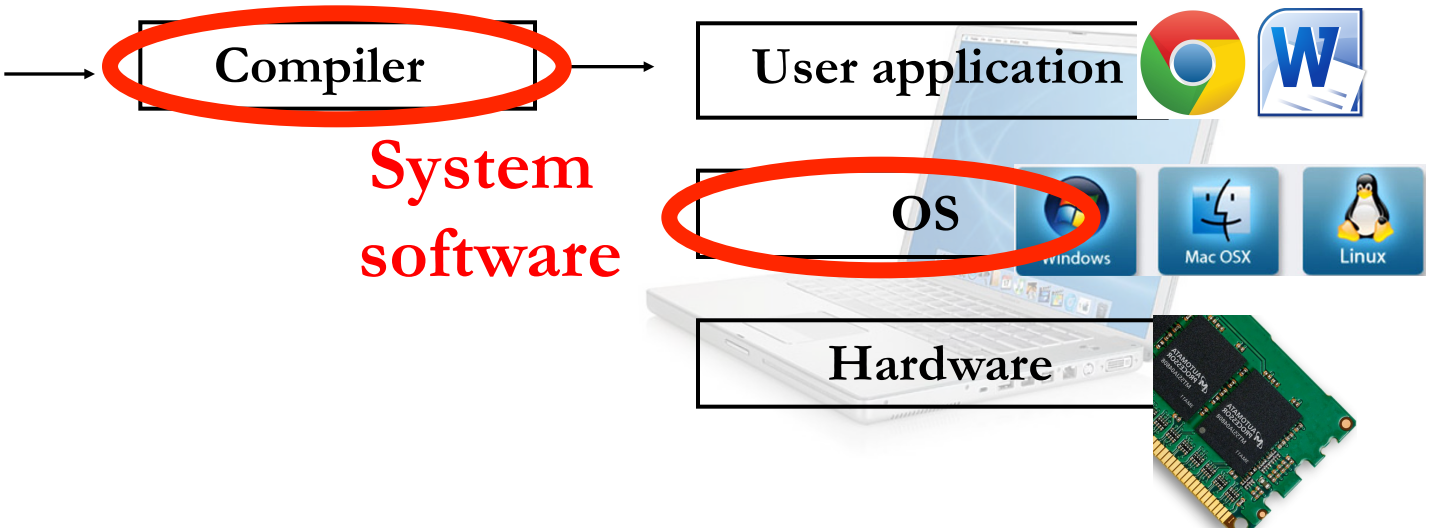
- Memory
  - Stores data and instructions
- Input/Output
  - Interfaces with other devices for getting/giving data



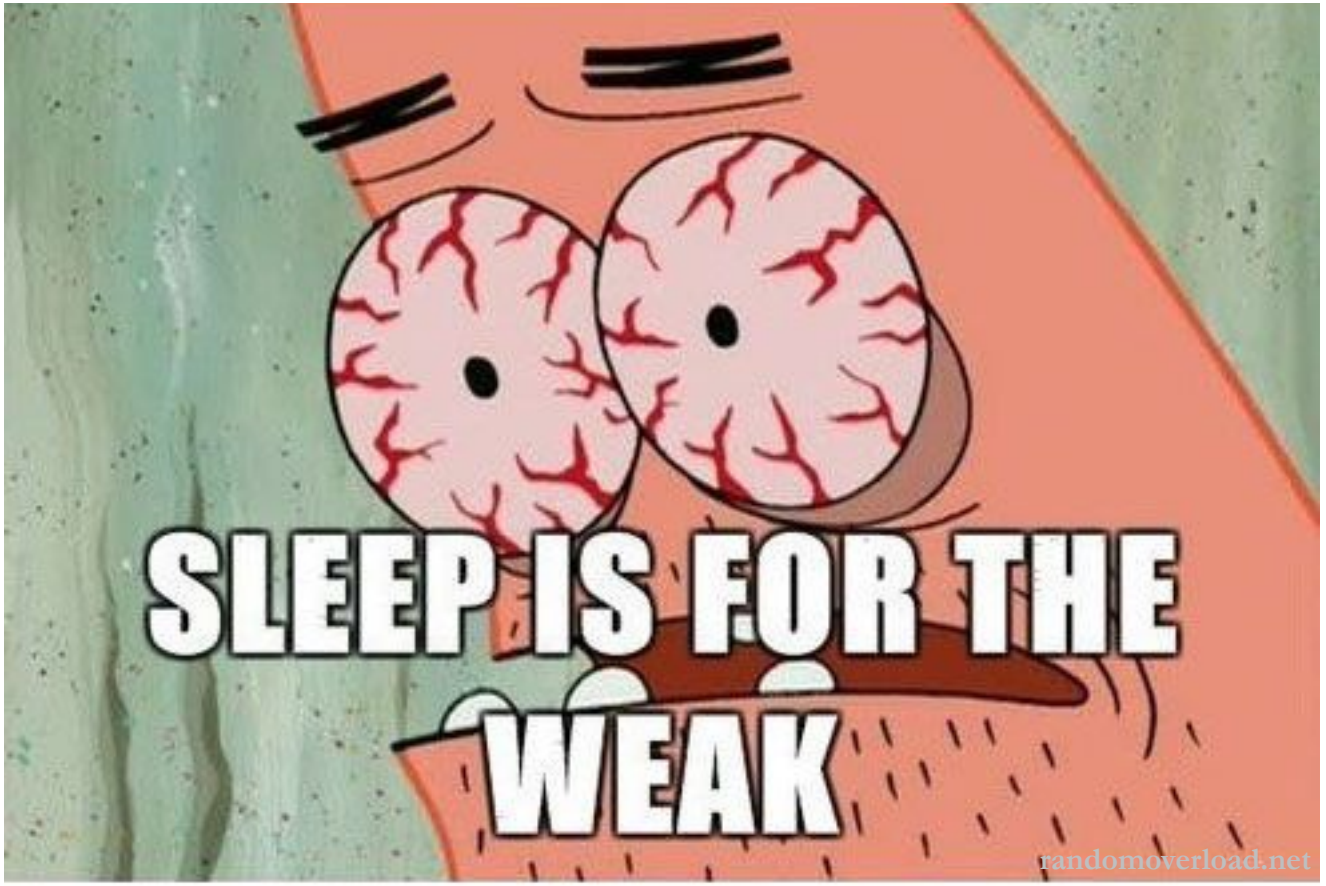
Modern computer system



Source code



- Compiler
  - Translates **High Level Language** (**HLL**) into **machine language** or **byte code**
- Operating System (OS)
  - Mediates access to hardware resources (CPU, Memory, I/O)
  - Schedules applications


SLEEP IS FOR THE
WEAK
randomoverload.net