



COMP 125: Introduction

Week 06, Midterm Review II



Review

1. History of Computers
2. Mainframes
3. Supercomputers
4. The PC
5. The Internet
6. Web browsers
7. Problem Solving Top-Down
8. Boolean Logic

History of Computers

- What were ancient computers?
- What did the industrial revolution mean for the advancements of computers?
- Bonus: name the very famous early binary machine
- How about WWII?
- What was the technology that meant computers could get both smaller and more capable?
- Why is binary important for computers?

Mainframes

- Describe the functionality of a mainframe.
- What is the “illusion” offered by the mainframe computer?
- Explain how the mainframe meant that the specializations of users could diverge from the specializations of computer administrators.
- How does a mainframe scale?
- When were mainframes popular? Are they still now?

Supercomputers

- Describe the functionality of a supercomputer.
- Describe how a supercomputer differs from a mainframe computer.
- How does a supercomputer scale?
- How have supercomputers changed over the years?

The PC

- Describe the functionality of the PC. Why were they so popular?
- How does a PC scale?
- How do PCs talk to each other?
- Explain how PCs are used differently today compared to 30 years ago.

The Internet

- How did the internet get started?
- Who initiated ARPANET and NSFNET? During which decade?
- What was the original purpose of ARPANET/NSFNET?
- When did the world wide web get started, and by whom?
- How did the WWW differ from ARPANET/NSFNET?
- What is the unit of data communication used in the internet?
- Describe how servers, packets, ISPs, and DNS servers all work together to get data from one place to another.

Web Browsers

- What is a web browser?
- How has the web browser changed in the course of the internet? Hint: where does the computation happen?
- What language are we using in this course, and what does it have to do with web browsers?

Problem Solving Top-Down

- Explain the top-down problem solving approach.
- What are the four things to look for when breaking the problem down?
 - Repetition (loops)
 - Variable
 - Function
 - None of the above? Make the steps smaller!

Op	Meaning
&&	AND (both True)
	OR (either is True)
!	NOT

```

a = 5
b = 10
c = -10
d = 10
e = 1
g = 2
h = 0

```

```

t = true
f = false

```

```

a + b
a * b
b ** a
d / g - e
d / (g - e)

```

```

a > b
b > c
(a > b) > g
(a > b) > c

```

```

h = h
h == h
g >= a
a <= c
d == c

```

```

h == f
e == t
t > f
t >= f
t > g
f > g

```

```

t && t
t || t
t || f
t && f
f && t
t && f ||
t
t && f ||
f
!t
!f
!t == f
!f > 0

```

Midterm

Monday: Written

- 10 problems
- Contextualization, history, or questions about what some piece of code is doing

Wednesday: Practical

- 3 lab-like problems
- Done in the editor

Friday: Oral

- During class period
- 5-7 minutes, discussion of contextualization or problem solving

Weakest of the three scores gets an automatic point boost (curved)