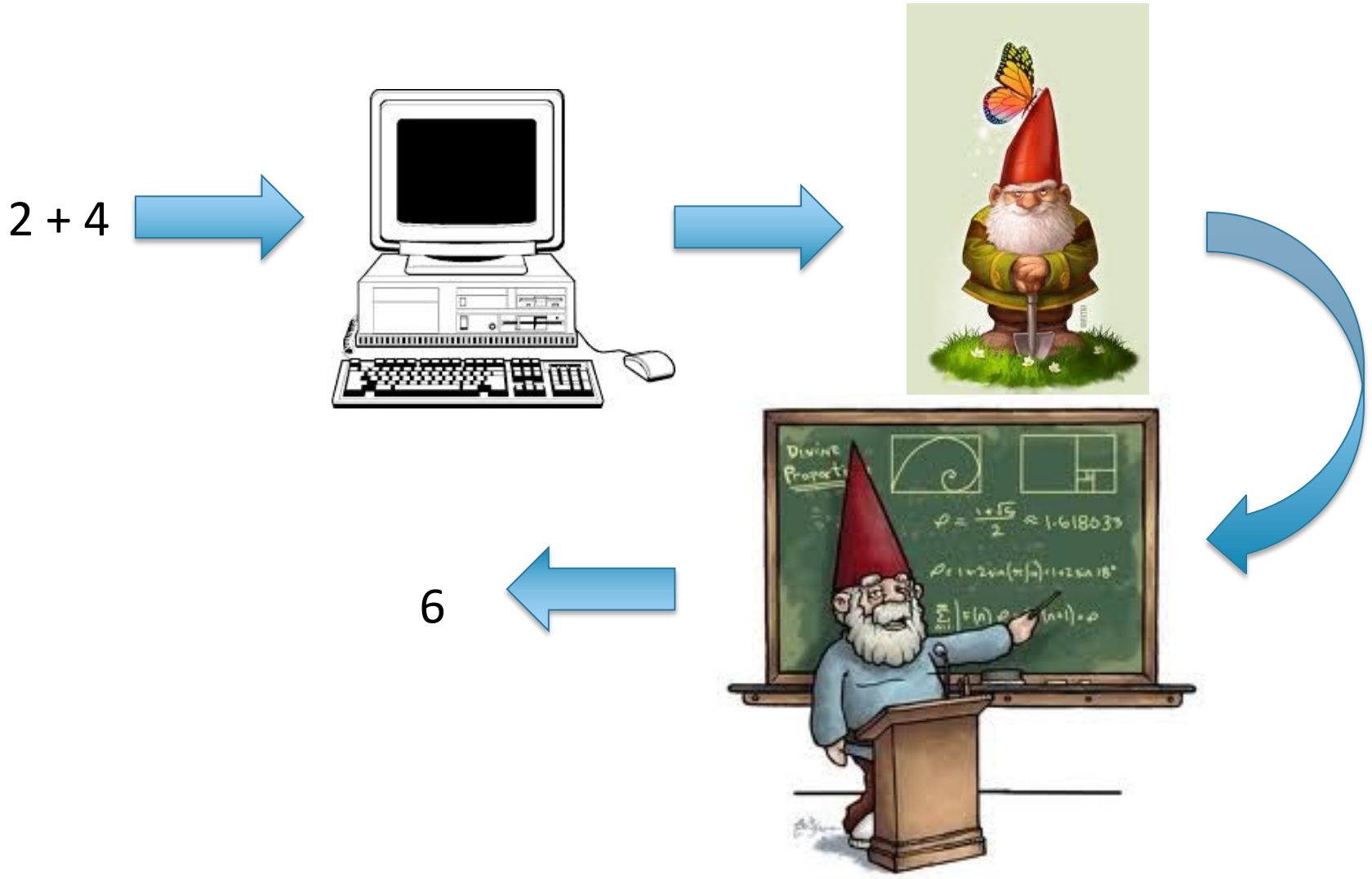
CSE 210: Computer Architecture
Lecture 1: Introduction

Stephen Checkoway
Oberlin College
Oct 4th, 2021
Slides by Cynthia Taylor

Announcements

- Office Hours:
  - Tuesday: 13:30–14:30
  - Friday: 13:30–14:30
- Course website: https://checkoway.net/teaching/cs210/2021-fall/
- Problem Set 0 due on Friday! It's on the website

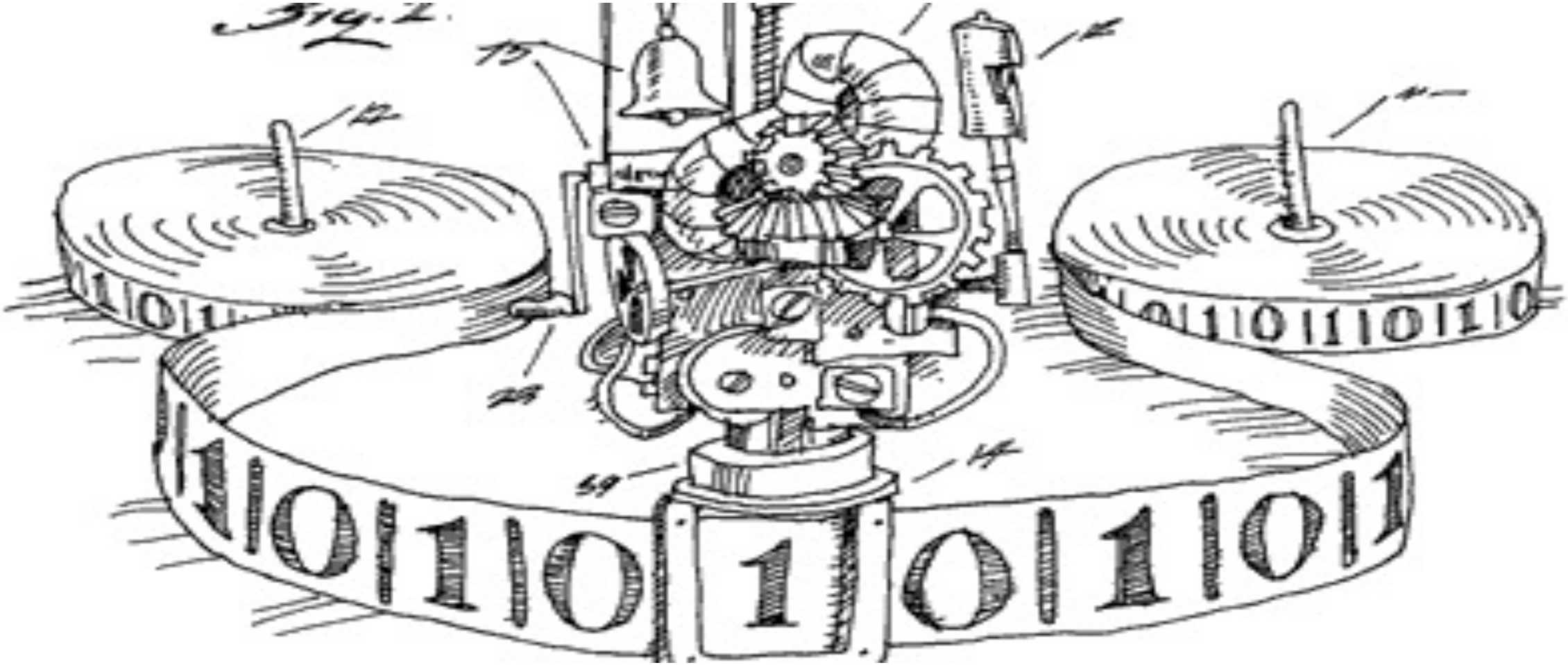
Previous Conceptions of How Computers Work



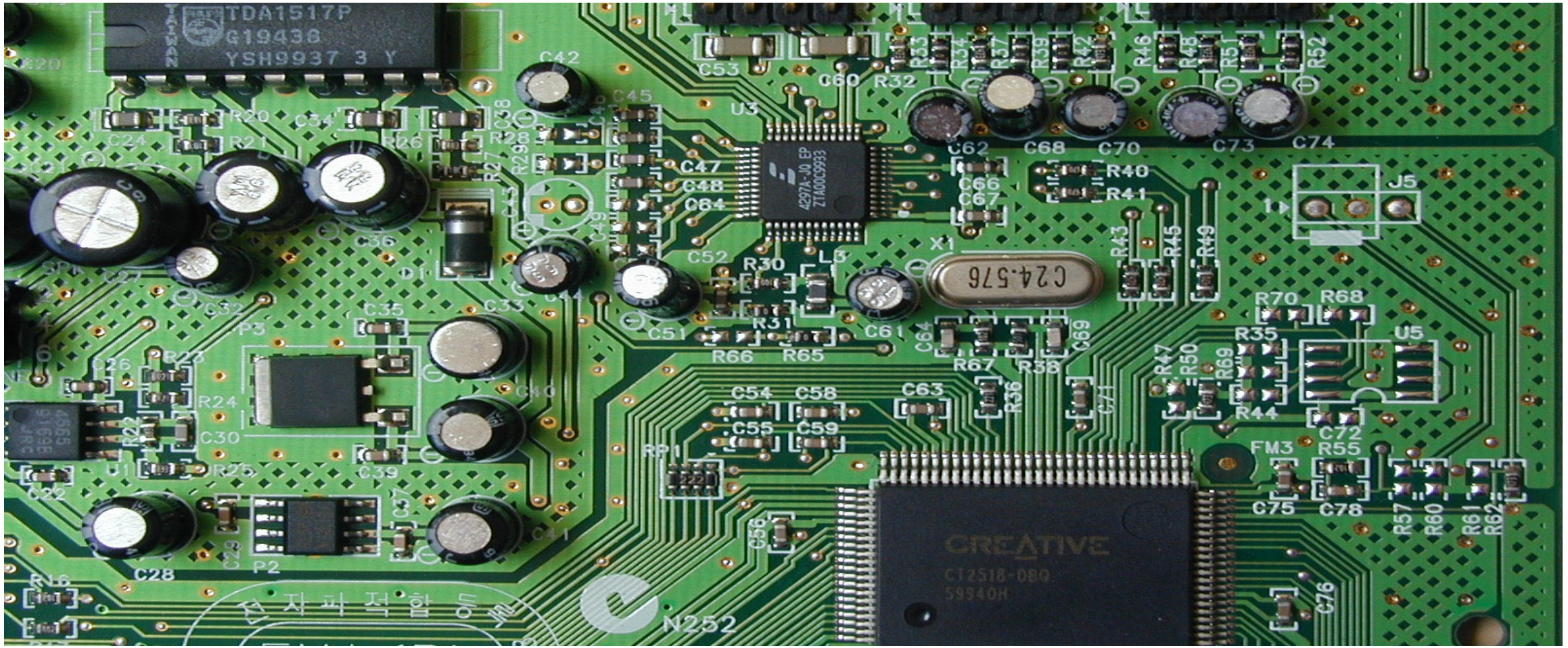
What is a computer?

What is a computer?

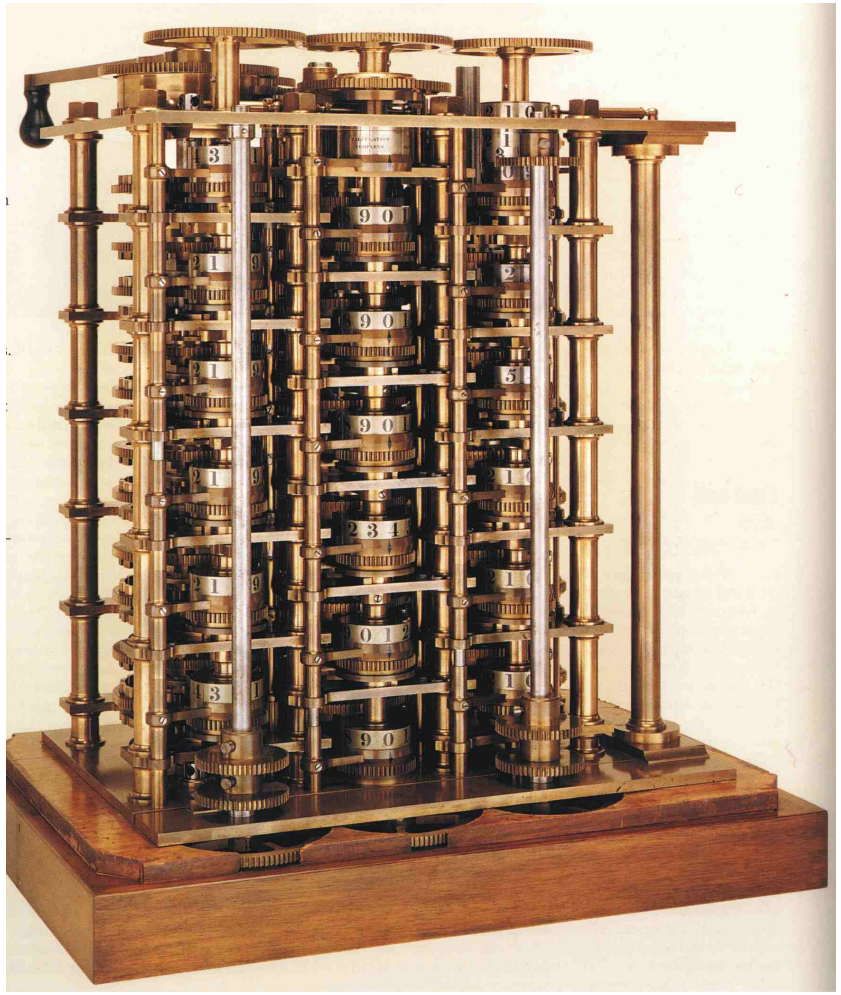
What is a computer?



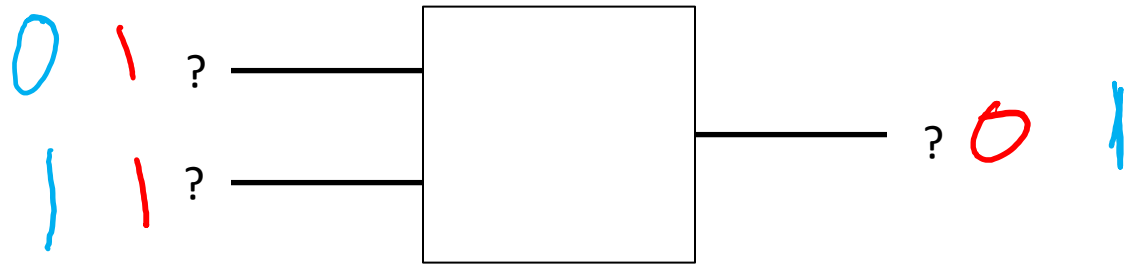
What is a computer?



Babbage's Difference Engine

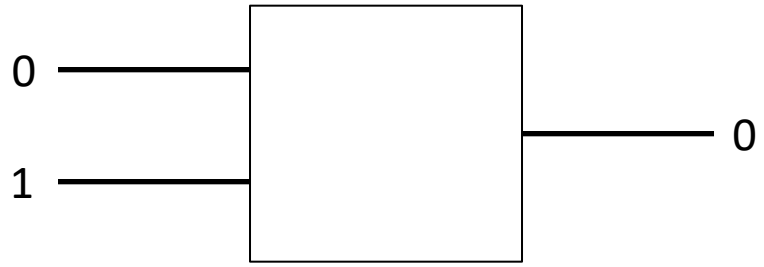


What is a computer?



- A device that reliably combines a given set of inputs to create the same output

But, that’s not a computer



```
def main():
    n = eval(input( "How many numbers should I sum?: "))

    sum = 0
    for i in range(1,n+1):
        sum = sum + i
    print("The sum of the first", i, "positive integers is", sum)

main()
```

Abstraction

- Non-metaphorical thought is only possible when we talk about purely physical reality.
  - George Lakoff, Metaphors We Live By

- What does this mean when we think about computers?

Discuss with your neighbors

- Introduce yourselves
- What are some different metaphors we use in computers?
- Non-metaphorical thought is only possible when we talk about purely physical reality.
  - George Lakoff, Metaphors We Live By
  - What does this mean when we think about computers?

Computer Metaphors

Levels of Abstraction

- User Interfaces
- High Level Languages
- Assembly Language
- Instruction Set Architectures
- Physical chip

In This Class

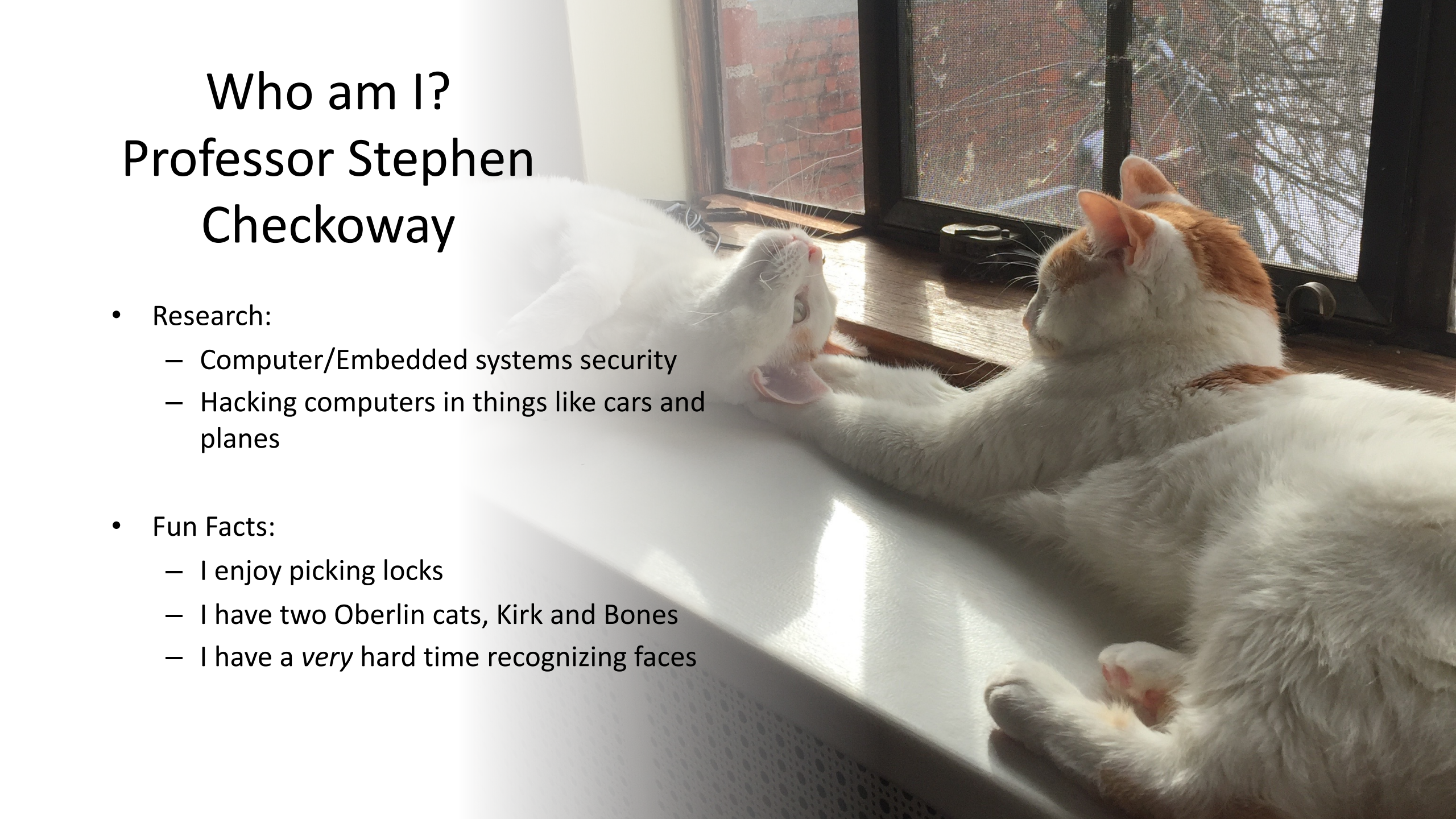
- What are the fundamentals we build these abstractions on top of?

- How do we create these abstractions?

Who am I? Professor Stephen Checkoway

- Research:
  - Computer/Embedded systems security
  - Hacking computers in things like cars and planes

- Fun Facts:
  - I enjoy picking locks
  - I have two Oberlin cats, Kirk and Bones
  - I have a *very* hard time recognizing faces



Class will be graded based on:

- Labs — Programming assignments
- Problem Sets — Written assignments
- Reading Exercises — From the zybook
- Class participation — Clicker questions!

Labs

- Programming assignments designed to explore the architecture concepts we learn in class
  - Java, MIPS, logic gates

- Due Sundays at 23:59

Problem Sets

- Written assignments where you solve problems related to computer architecture

- Examples:
  - Converting numbers to binary or hex
  - Simple MIPs programs
  - Drawing circuit diagrams
  - Answering questions about the datapath

Problem Sets

- Can be resubmitted within 1 week of receiving your grade
- Final problem set grade is 25% your original submission grade, 75% your new grade.
- Due Fridays at 23:59
- Problem Set 0 due this Friday!

Reading

- We will be using a zybook
- Can buy directly from zybook, or buy an access code from the bookstore
  - Sign in or create an account at learn.zybooks.com.
  - Enter zybook code: `OBERLINCSCI210CheckowayFall2021`
  - Subscribe
- If you cannot afford the textbook, Oberlin's Emergency Textbook fund can offer you a loan due only when you are financially able

Reading

- Due BEFORE CLASS on the day it is listed on the class schedule

- https://checkoway.net/teaching/cs210/2021-fall/schedule.html

Clickers!

- Lets you vote on multiple choice questions in real time.
- Like pub trivia, except the subject is always computer architecture.
- You need one by next Monday

Group Discussion Norms

- Make sure everyone gets to talk.
- Have everyone state their answer before discussing which answer is correct.
- Take turns reporting out.
- If you think someone is wrong, ask them to explain their thinking rather than just dismissing it.

Class Norms

- Contribute as you feel comfortable
  - If you're not comfortable answering, you can pass.
  - If you're not usually inclined to speak much in class, push yourself to ask questions more often.
- Be aware of the space you take up in class
  - Make space for others, use some space for yourself
- The main goal of every person in the class should be to engage proactively with the ideas we understand the least. If someone asks a question/makes a comment that seems obvious to you, show them respect.

Collaboration Policy

- Discuss the labs/problem sets with anyone
- Post questions on piazza

- Don’t show anyone your code
- If you work through how to solve a problem, please change relevant numbers from the assigned problems
- Must write down answers separately

The Challenge of Computer Architecture

- The industry changes faster than any other.

- The ground rules change every year.
  - new problems
  - new opportunities
  - different tradeoffs

- It’s all about making programs run faster or use less energy or provide more features than the other company’s machine.

Understanding Computer Architecture Will Let You

- Write better code
- Write faster code
- Understand what is and isn’t possible

Reading

- Next lecture: Assembly Language
  - Read zybook Section 1.3 & 1.4

- Problem Set 0 due Friday 23:59