

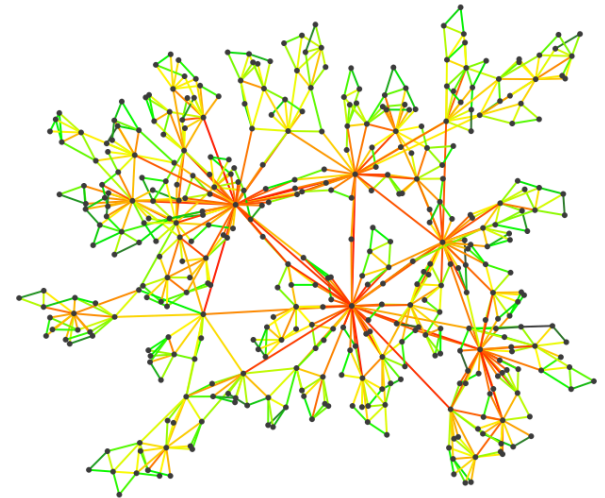
WHAT IS PROGRAMMING?

CS302 – Introduction to Programming
University of Wisconsin – Madison
Lecture 1

By Matthew Bernstein – matthewb@cs.wisc.edu

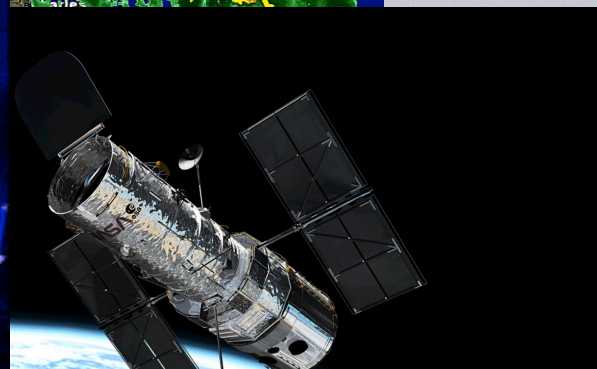
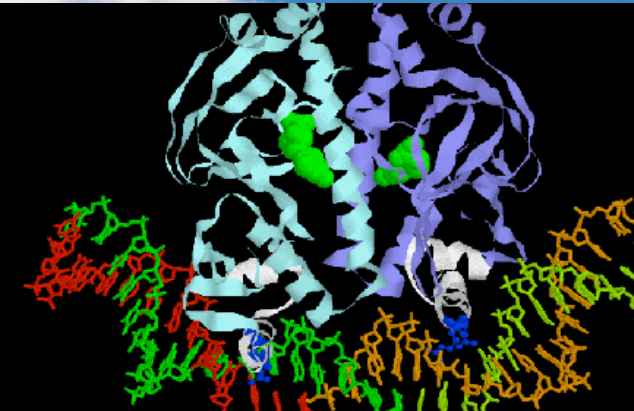
What is Computer Science?

- Computer Science is the study and application of computation and automation
- It can be very theoretical and abstract:
 - Computational Complexity Theory
 - Formal Language Theory
 - Information Theory
- It can be very applied:
 - Software development
 - Computer Graphics
 - Artificial Intelligence



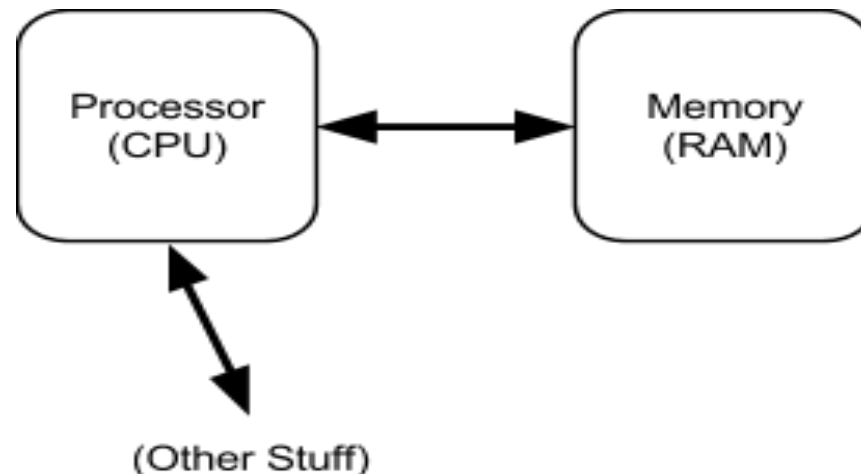
Applications of Computer Science

- **Artificial Intelligence:** Finance, Robotics, Biometrics, Human Speech, E-Commerce
- **Computer Graphics:** Video Games, Simulators, Movies, User Interfaces
- **Networks:** Internet, Biological Networks, Social Networks
- **Systems:** Operating Systems, Distributed Systems, Databases
- **Biology and Medicine:** Bioinformatics, Computational Medicine
- **Software Engineering:** Mobile Apps, Internet Apps



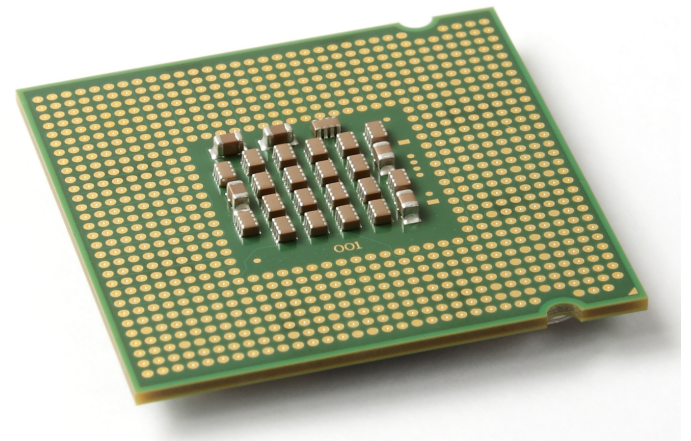
The Computer

- A programmable tool for doing fast computation
- Three central components:
 - **CPU** (The brain, does the computation)
 - **Memory** (Stores the results of the computation and the instructions needed to do the computation)
 - **Devices** (Monitor, Mouse, Hard Drive, etc.)



The Central Processing Unit (CPU)

- Composed of several hundred million small electronic components called transistors
- Executes very simple, specific instructions:
 - Add numbers
 - Multiply numbers
 - Store numbers in memory
 - Etc.
- These instructions are coded in a machine language (tedious to write and difficult to understand by humans)



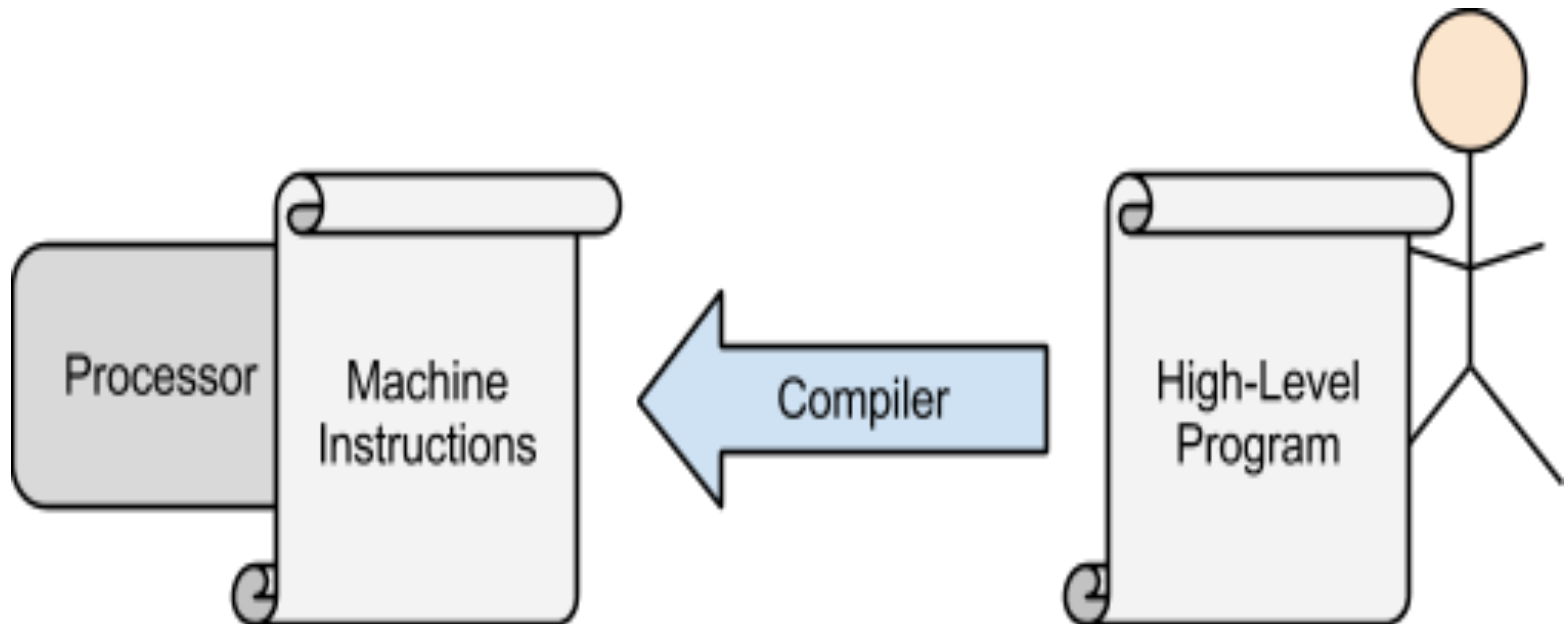
What is Programming?

- Programming is the process of designing and building sets of instructions that a computer can understand and execute

Programming Languages

- Humans write instructions for a computer using a programming language
 - Written onto plain old text files
- Programming languages are much easier for humans to deal with than machine languages
- A **compiler** is a program that translates the programming language into machine compatible instructions (a.k.a. machine language)

Basically...



Java

- Java is one of many programming languages
- Compiler translates Java code to **Java bytecode**
- Java bytecode is a set of instructions that a **Java Virtual Machine (JVM)** can use to run instructions on your CPU
- Compilation:
 - Java Source → Compiler → Bytecode → JVM
- JVMs allow for Java code to run on many different types of computers consistently

Algorithms

- A step-by-step procedure used for completing a task
- Examples:
 - Recipes for cooking
 - Directions for getting from place to place
- Computers follow extremely detailed algorithms
 - Must be absolutely unambiguous
 - A computer will not fill in any gaps. The computer is a tool that operates on a certain set of possible instructions.
- As a programmer, you provide the algorithms via a programming language

Writing Java Code

- Java is written on plain text files (with a “.java” extension)
- The .java extension tells other programs (such as the compiler) what kind of data is stored in the file (a.k.a. Java code)
- The files that contain the compiled bytecode have a “.class” extension (don’t mess with these)
- There are tools that help you write Java... namely...Eclipse!

Using Eclipse to Help You Program

- Eclipse is a type of **Integrated Development Environment (IDE)**
- It is basically just a big program that helps you build new programs by:
 - Allowing you to edit the source files
 - Highlighting keywords in your programs so you can easily see the structure of your program
 - Compiling your program and telling you if you made a mistake
 - Providing tools for debugging



Algorithm Design

- Knowing how to use a programming language is great, but if you don't know how to design algorithms, your programming will greatly suffer
- On the contrary, if you are a good algorithm designer, picking up a programming language is relatively easy.
- Analogy: Knowing English, but being a poor writer and trying to write a novel

Writing Pseudocode

- **Pseudocode** is an informal description of an algorithm
 - It does not necessarily have to be written in a syntactically correct programming language.
- Pseudocode for finding the best quality photo amongst many photos:
 1. Pick the first photo and call it “the best so far”.
 2. For each photo in the sequence:
 3. If it is more attractive than the “best so far”:
 4. Discard the best so far
 5. Call this photo “the best so far”
 6. The photo called “the best so far” is the most attractive photo in the sequence

Cool CS Link of the Day

Code.org's video on why everyone should learn computer programming:

<http://www.youtube.com/watch?v=nKlu9yen5nc>

“Everybody in this country should learn how to program a computer... because it teaches you how to think.”

- Steve Jobs