



# **Introduction to Programming**

Spring 2022

# Objectives

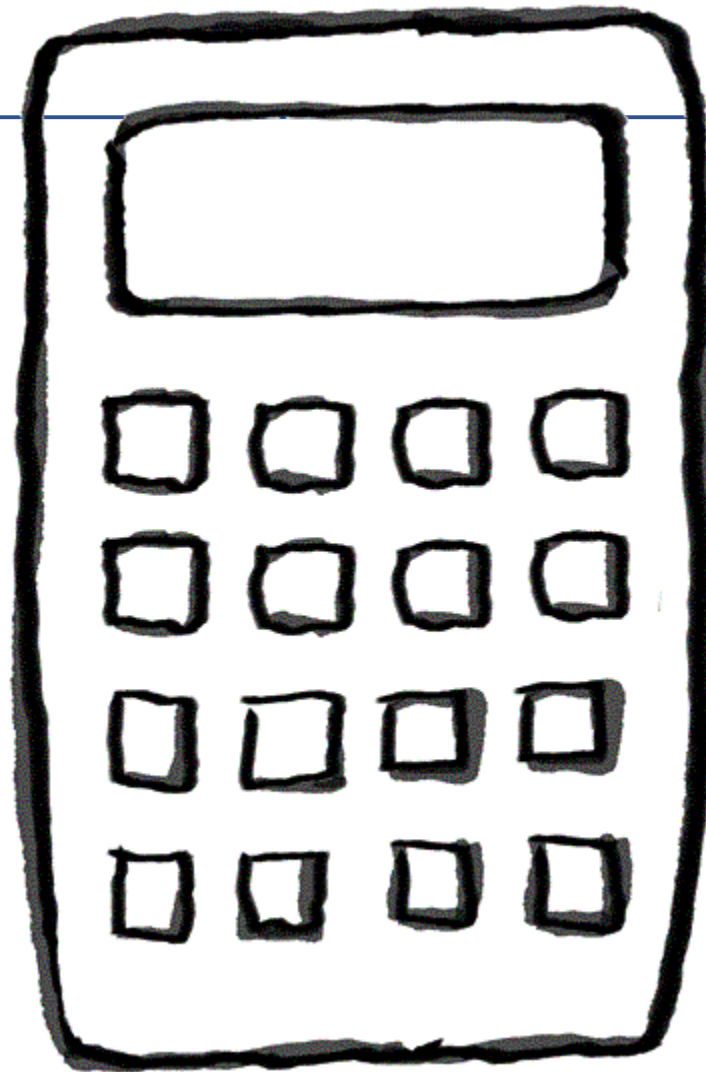
---

- Computer Programming
- Algorithm
- The Programming Process
- Programming Languages
- Installing Python and IDLE

# Computer Programming

---

- Aspects of a computer program that must be designed:
  - The logical flow of the instructions
- Step by step set of instructions
  - The way information is presented to the user
  - The program's "user friendliness"
  - Manuals, help systems, and/or other forms of written documentation?





# Computer Programming

---

- Programs must be analytically correct as well.
- Programs rarely work the first time they are programmed.



# Computer Programming

---

- Programming Process
  - Design
  - Analysis
  - Experimentation

# Design

---

- One way to show a particular problem can be solved is to actually design a solution.
- This is done by developing an algorithm, a step-by-step process for achieving the desired result.
- One problem – it can only answer in the positive. You can't prove a negative!

# Analysis

---

- Analysis is the process of examining algorithms and problems mathematically.
- Some seemingly simple problems are not solvable by any algorithm. These problems are said to be unsolvable.
- Problems can be intractable if they would take too long or take too much memory to be of practical value.
- Computer Scientists talk about P vs NP computational complexity



# Experimentation

---

- Some problems are too complex for analysis.
- Implement a system and then study its behavior.



# Computer Programming

---

- Programming languages have strict rules, known as syntax, that must be carefully followed.

# Computer Programming

---

- Syntax
  - Rules of a language
- English:
  - Subject Verb Object
  - Cat boy hugs
  - Cat hugs boy

# Computer Programming

---

- Syntax
  - Rules of a language
- Programming Language:
  - Language depended
  - $3.2 + 10$
  - $3.2, 10, +$
  - "High Five"

# Computer Programming

---

- Semantics

- Meaning associated with a syntactically correct string of symbols

- English:

- “Flying planes can be dangerous”

- “This reading lamp hasn’t uttered a word since I bought it?”

# Computer Programming

---

- Programming languages have only one meaning but may not be what programmer intended
  - Programmer wants to add two numbers (5 and 3) but wrote:  
– 5 - 3

# Computer Programming

---

- Syntactic errors
  - Common and easily caught
- No syntactic errors but different meaning than what programmer intended (semantic errors)
  - Program crashes, stops running
  - Program runs forever
  - Program gives an answer but different than expected

# Computer Programming

---

- A program is a set of instructions a computer follows in order to perform a task.
- A programming language is a special language used to write computer programs.
- Collectively, these instructions form an algorithm



# Algorithm

---

- An algorithm is a set of well defined steps to completing a task.
- The steps in an algorithm are performed sequentially.
- Algorithms are like recipes

# Algorithm

---

- An algorithm may be:
  - Incorrect
  - Correct but inefficient
  - May never end
- Algorithm must be precise
- Algorithms are written in pseudocode

# Example of an Algorithm

---

## .Cooking Rice

- Measure one cup of rice
- Measure two cups of water
- Pour water into a pot
- Put the pot on the stove
- Turn on the stove setting it to high
- Boil the water
- Add the rice to pot
- Turn down the heat to simmer
- Cover the pot
- Cook for twenty minutes
- Turn off the heat and remove the lid

# Example of an Algorithm

---

- Cooking Rice
  - Add the Rice
  - Turn Off the Heat and Remove the Lid
  - Measure the Rice and Water
  - Boil the Water
  - Cover and Cook till Rice is done

# Example of an Algorithm

---

- Driving Direction from Denver to Colorado Springs
  - Take I-25 North
  - Take I-70 East
  - Take I-225 South
  - Take I-25 South

# Example of an Algorithm

---

- Write an algorithm using pseudocode to calculate gross pay.
  - Get payroll data.
  - Calculate gross pay.
  - Display gross pay.

# Example of an Algorithm

---

- Write an algorithm using pseudocode to calculate gross pay.
  - Display “How many hours did you work?”
  - Input hours.
  - Display “How much do you get paid per hour?”
  - Input rate.
  - Store the value of hours times rate in the pay variable ( $\text{pay} = \text{hours} * \text{rate}$ ).
  - Display the value in the pay variable.

# Stages in Software Development

---

- Process for software development can be divided into five steps:
  - Writing specifications
  - Developing solution or algorithms
  - Coding algorithms
  - Testing
  - running



# Writing Specifications

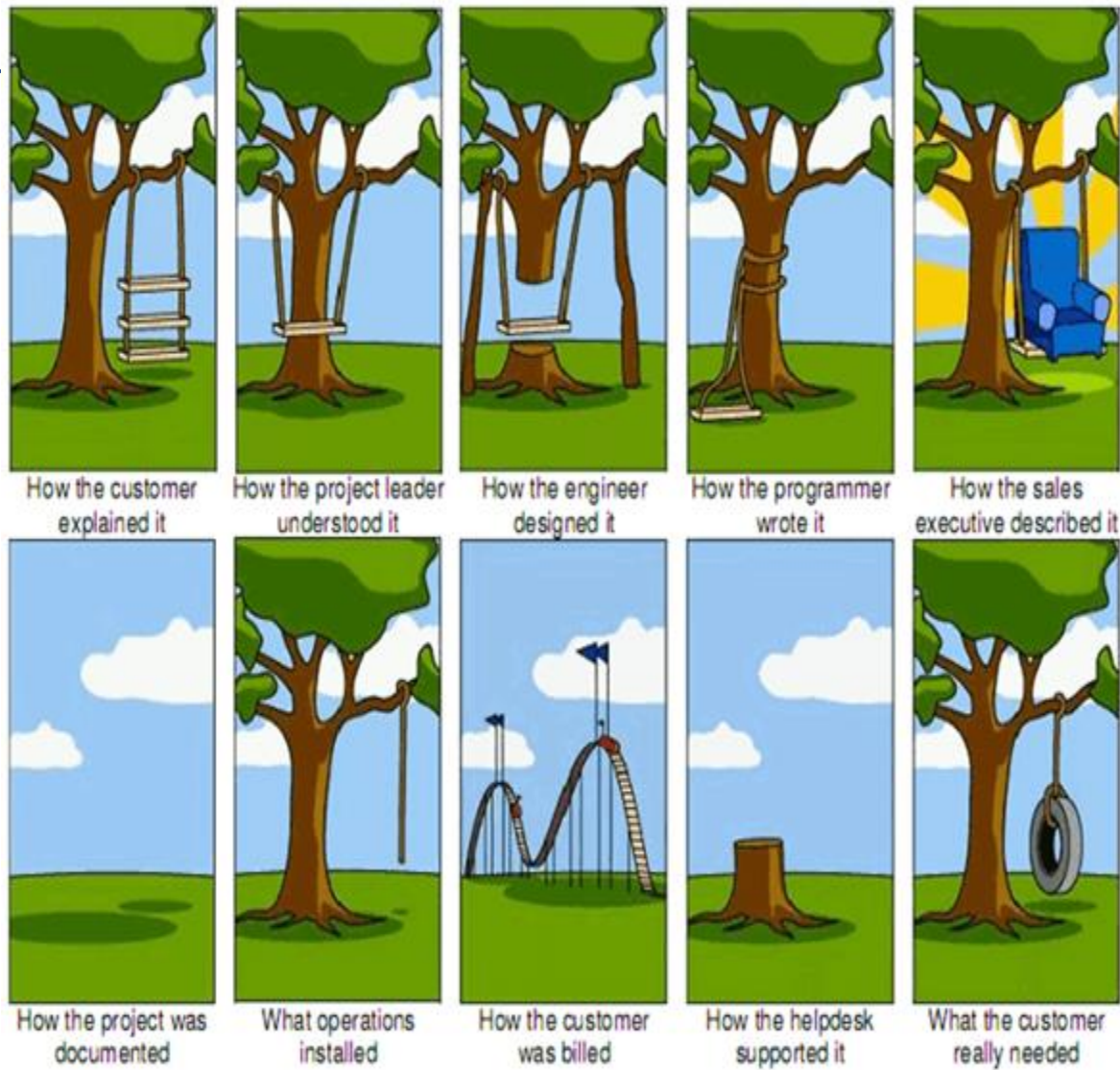
---

- Describe the problem to be solved
- It can start with a concept
- Specifications can be vague
- Where do we get the specifications?
  - From users
  - Users might not know all the possibilities

# Requirement Gathering

---

- Users might think we can do anything:
- <https://www.youtube.com/watch?v=BKorP55Aqvg>



# Writing Specifications

---

- Bulleted list of functionalities included in the system

# Programming Languages

---

- A computer needs the algorithm to be written in machine language.
- Machine language is written using binary numbers.

# Programming Languages

---

- The binary numbering system (base 2) only has two digits (0 and 1).
- The binary numbers are encoded as a machine language.
- Each CPU has its own machine language.
  - Motorola 68000 series processors
  - Intel x86 series processors
  - ARM processors, etc.
- Example: 10110100000000101

# Programming Languages

---

- In the distant past, programmers wrote programs in machine language.
- Programmers developed higher level programming languages to make things easier.
- The first of these was assembler.
- Assembler made things easier but was also processor dependent.

# Programming Languages

- High level programming languages followed that were not processor dependent.
- Some common programming languages:

Java	C	Visual Basic
BASIC	C++	Python
COBOL	C#	Ruby
Pascal	PHP	JavaScript