ITEC 5920W
Applied Programming

Introductory Lecture
Computers and Programs
Jan 7, 2020

Today’s Outline

- Course Outline
- Computers and Programs
- Basic Python Programming

Instructor

Instructor: Emma Farago
Email: emmafarago@cmail.carleton.ca
Office: Canal Building 7207
Office Hours: Set by appointment

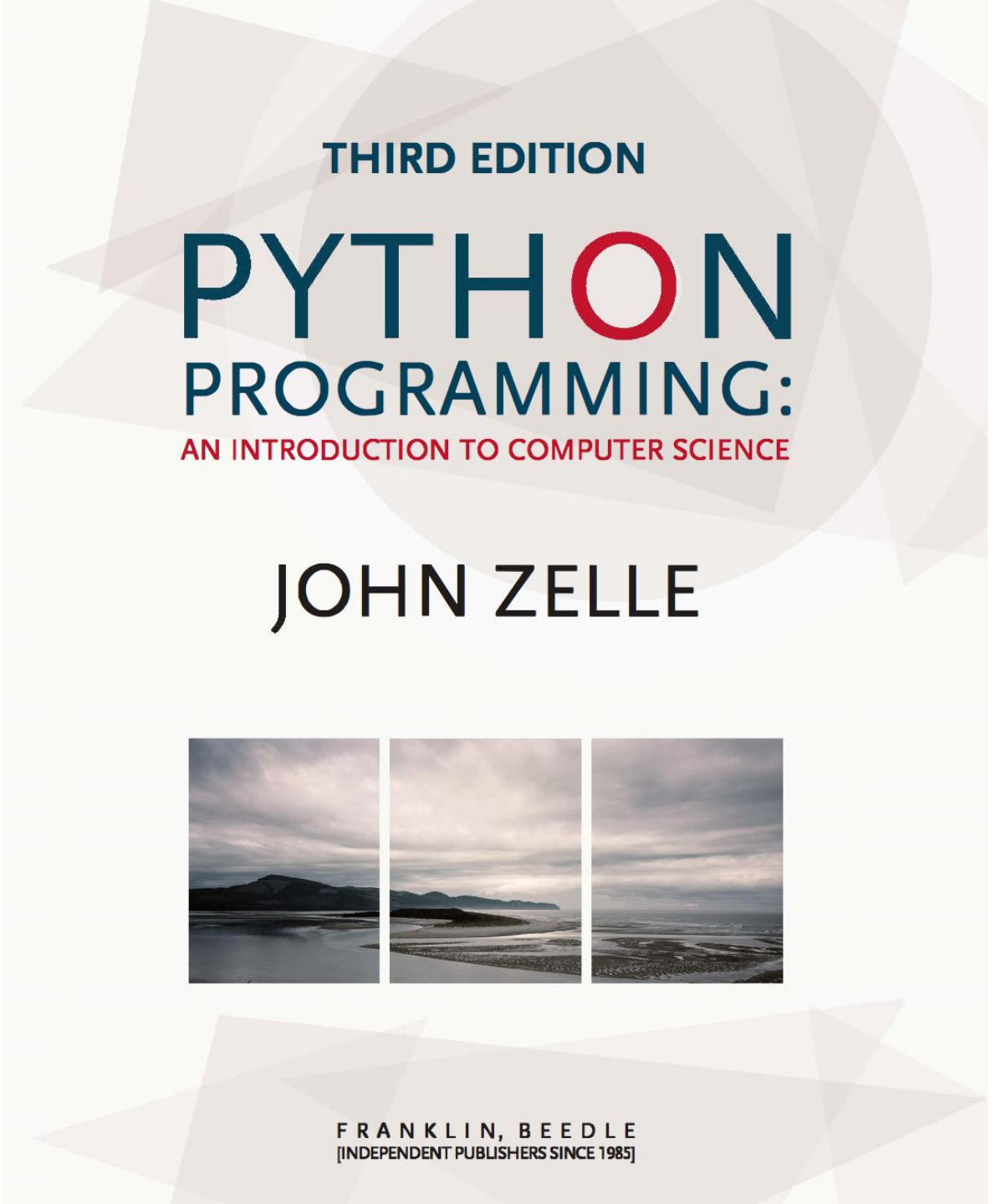
Textbook

Python Programming:

An Introduction to Computer Science

Third Edition

by John Zelle



Course Topics

- Programming fundamentals
- Object-oriented programming
- Data collections
- Algorithm Design
- Special topics - let me know if there is anything you want to cover in class

Evaluations

- Weekly Quizzes (10%)
  - posted on Wednesdays
  - due on Sundays at 11:55 pm
- Labs and Tutorial Assignments (20%)
  - Weekly tutorials on Fridays 8:35-11:35am
- Midterm (30%)
  - Friday Feb 28 (in tutorial)
- Individual Final Project (40%)
  - Proposal due Friday March 6

Computers and Programs

https://www.youtube.com/watch?v=xNjgSKGqjno

Computers and Programs

https://www.youtube.com/watch?v=xNjgSKGqjno

“I’m designing a machine that will allow us to break every message every day instantly.”

Computers and Programs

https://www.youtube.com/watch?v=xNjgSKGqjno

Computers and Programs

https://www.youtube.com/watch?v=xNjgSKGqjno

Computer science: study of what processes can be described by a computer, and what can be computed

Intractable problem: a problem that is possible to solve, but would take too long or require too much computer memory to practically solve

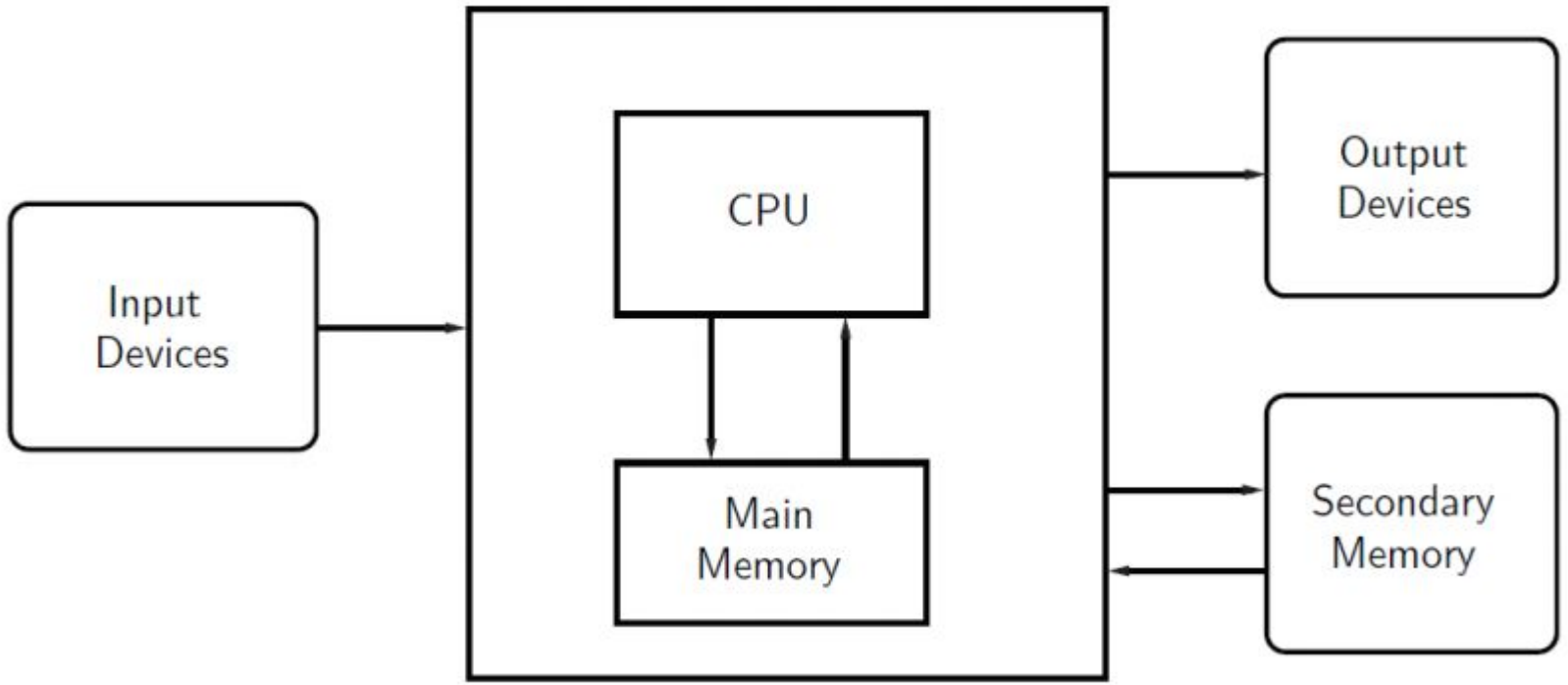
ex. prime factorization

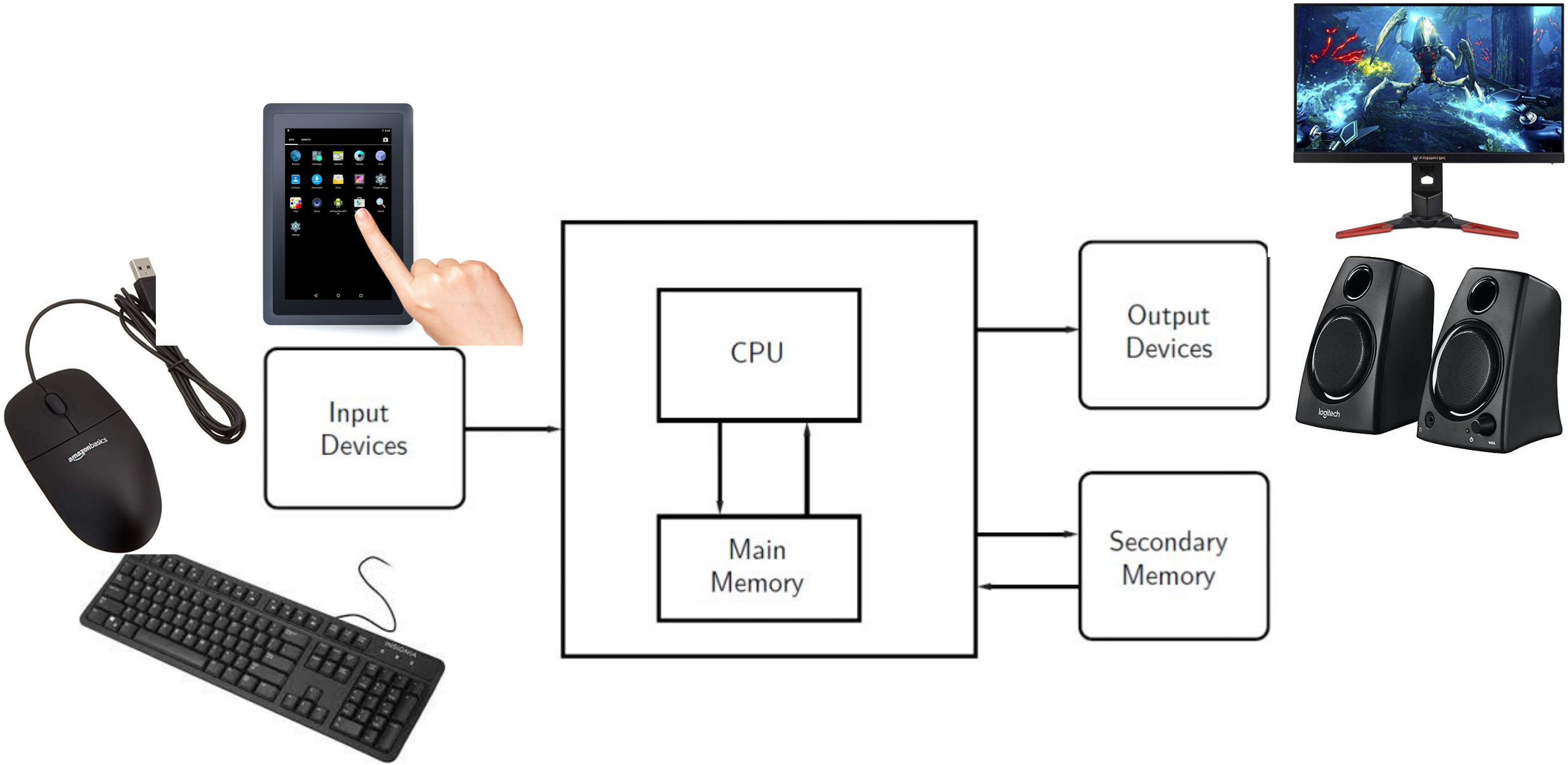
Programming instructions

- A computer program is a **list of instructions** of the operations that the computer should perform

- Examples of instructions:
  - save information in memory
  - retrieve information from memory
  - add two numbers
  - test if two numbers are equal

Computer Hardware




Input Devices
CPU
Main Memory
Output Devices
Secondary Memory

Central processing unit (CPU)

- “Brain” of the computer
- Controls flow of information within the computer and executes instructions
- Contains Arithmetic Logic Unit (ALU)
- General purpose (can perform many different types of calculations

Clock Rate

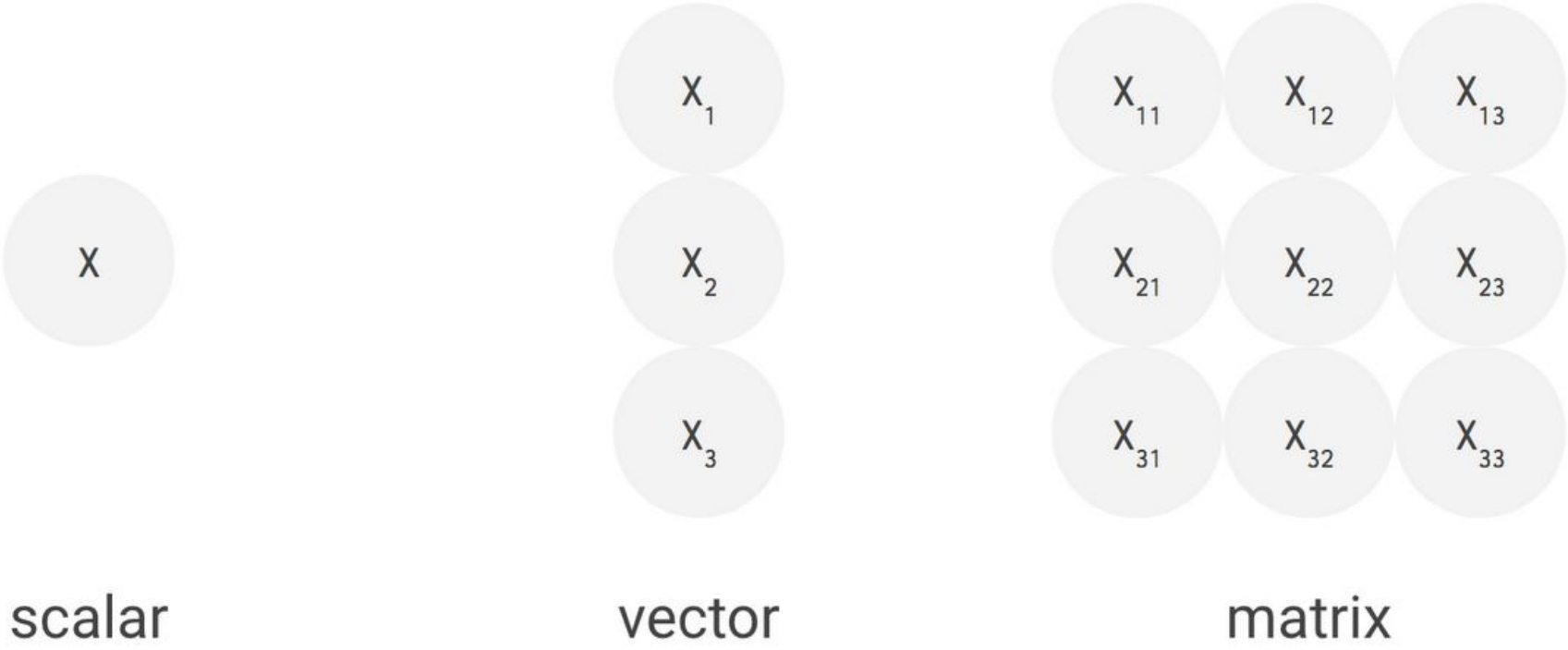
- Synchronizes the operation of processor components
- Sometimes* helps indicate the speed of a processor
- Typical CPU performs one scalar operation with every clock cycle
- Typical computer is in the GHz range (10^9 clock cycles per second)

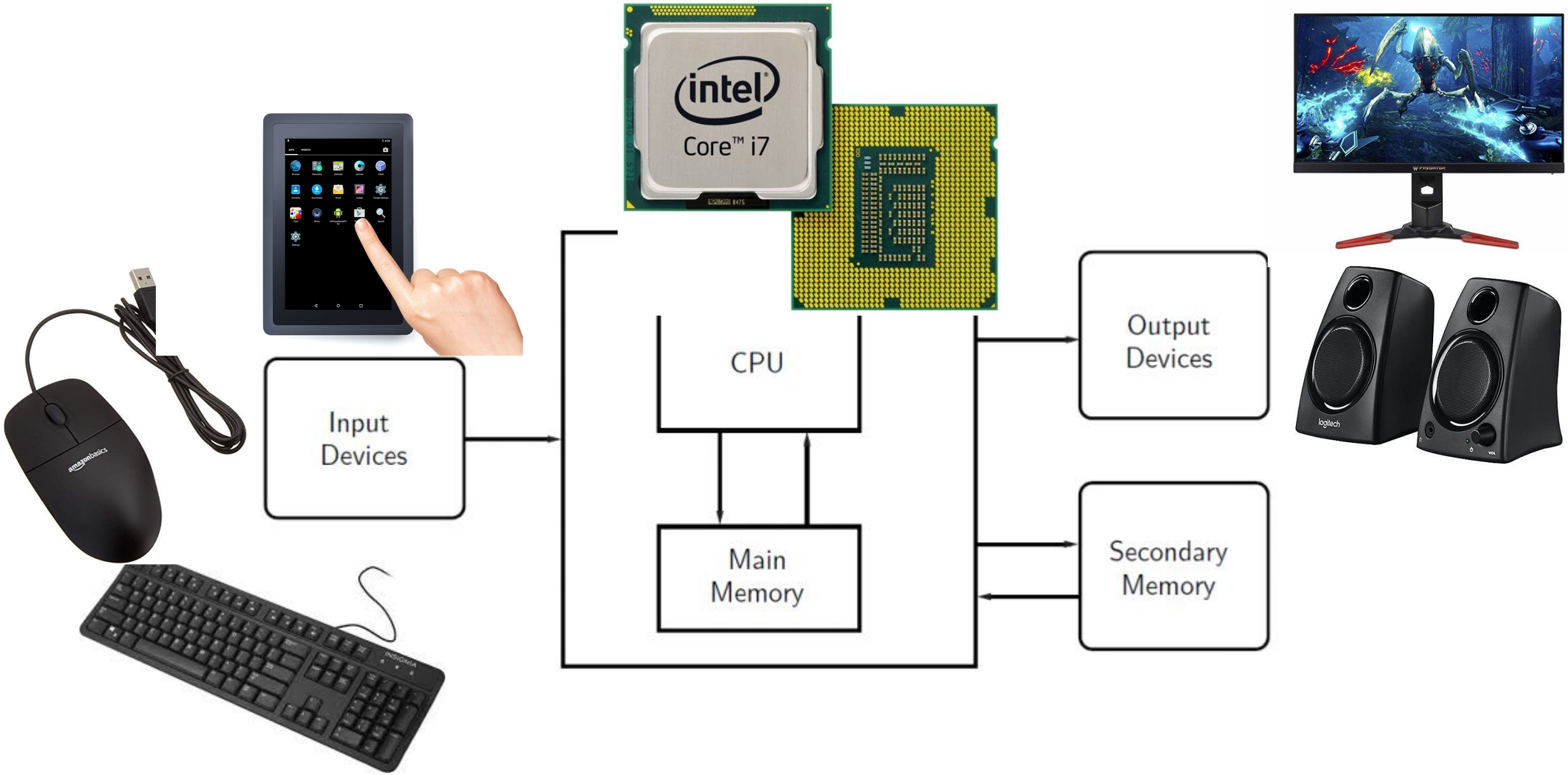
Graphics Processing Unit (GPU)

- Efficient at image processing
- Can perform an operation on thousands of data elements per instruction/clock cycle by using vector processing

Tensor Processing Unit (TPU)

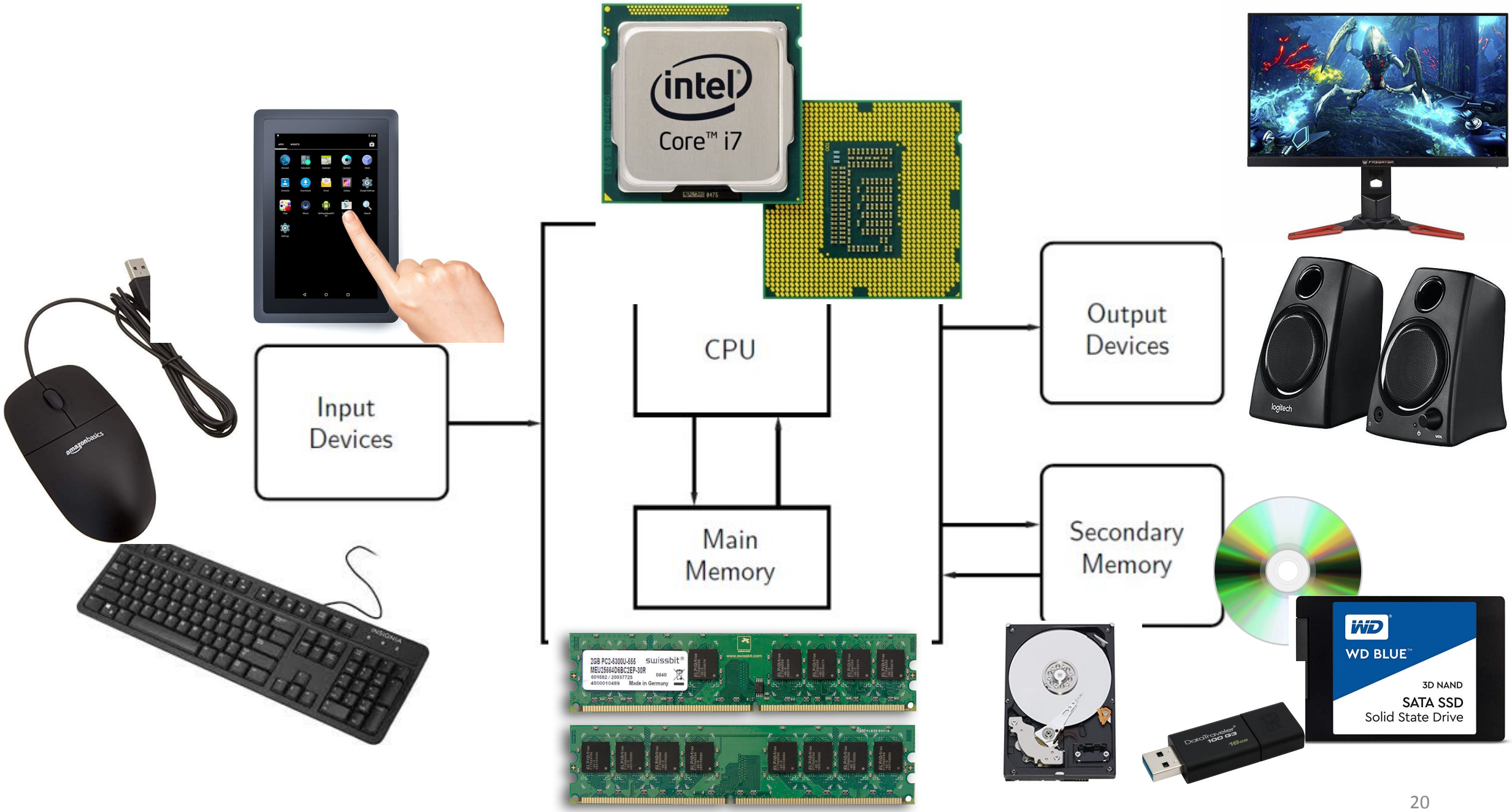
- Hardware accelerator developed by Google for artificial intelligence (AI) and deep learning
- Efficient at matrix multiplication




Input Devices
CPU
Main Memory
Output Devices
Secondary Memory

Memory

- The CPU retrieves and executes instructions stored in the computer memory
- Main memory: directly accessed by the CPU
  - primarily Random Access Memory (RAM)
  - RAM is volatile (lost when power is turned off)
- Secondary memory: long(er)-term data storage
  - hard drive (magnetic)
  - Solid state drives (flash)
  - USB stick (flash)
  - CD/DVD (optical)


Input Devices
CPU
Main Memory
Output Devices
Secondary Memory
intel
Core™ i7
WD BLUE™
3D NAND
SATA SSD
Solid State Drive

Programming instructions

- Recall that a computer program is a **list of instructions**

Programming Languages

- CPUs only understand machine code
- Each CPU has an instruction set
- Each instruction is a binary pattern (0s and 1s) that corresponds to a command

  ex: 10110000 00000001
  10110001 00000010
  00000000 01011011

Programming Languages

- CPUs only understand machine code
- Each CPU has an instruction set
- Each instruction is a binary pattern (0s and 1s) that corresponds to a command

  ex: 10110000 00000001
  10110001 00000010
  00000000 01011011

- Assembly language represents the binary instructions as more readable instructions
  - ex. MOV, DEC, ADD, SUB, JMP

Programming Languages

MOV AL, 1h
MOV CL, 2h
ADD DL, 5Bh

10110000 00000001
10110001 00000010
00000010 01011011

Programming Languages

- High-level programming languages are even easier to program
- Compiler is used to translate Python code to machine code

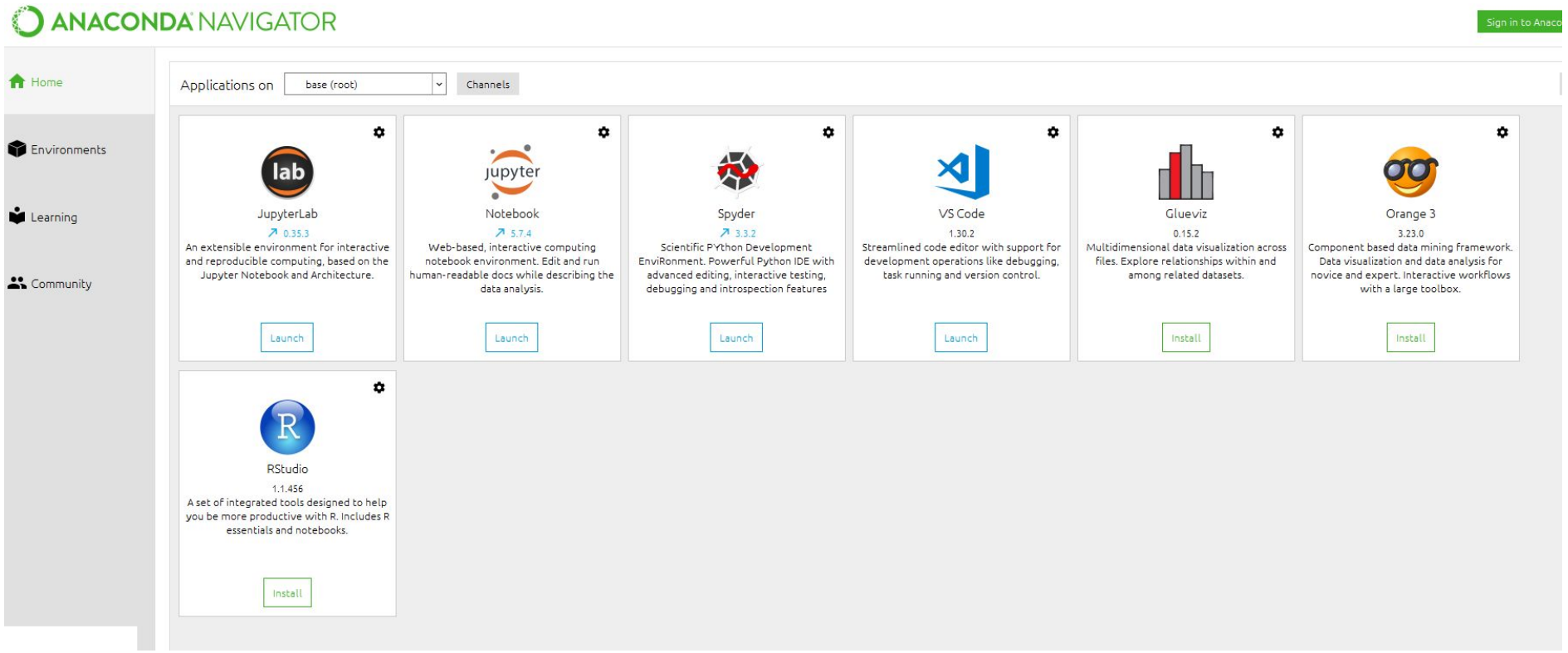
Compiling vs. Interpreting

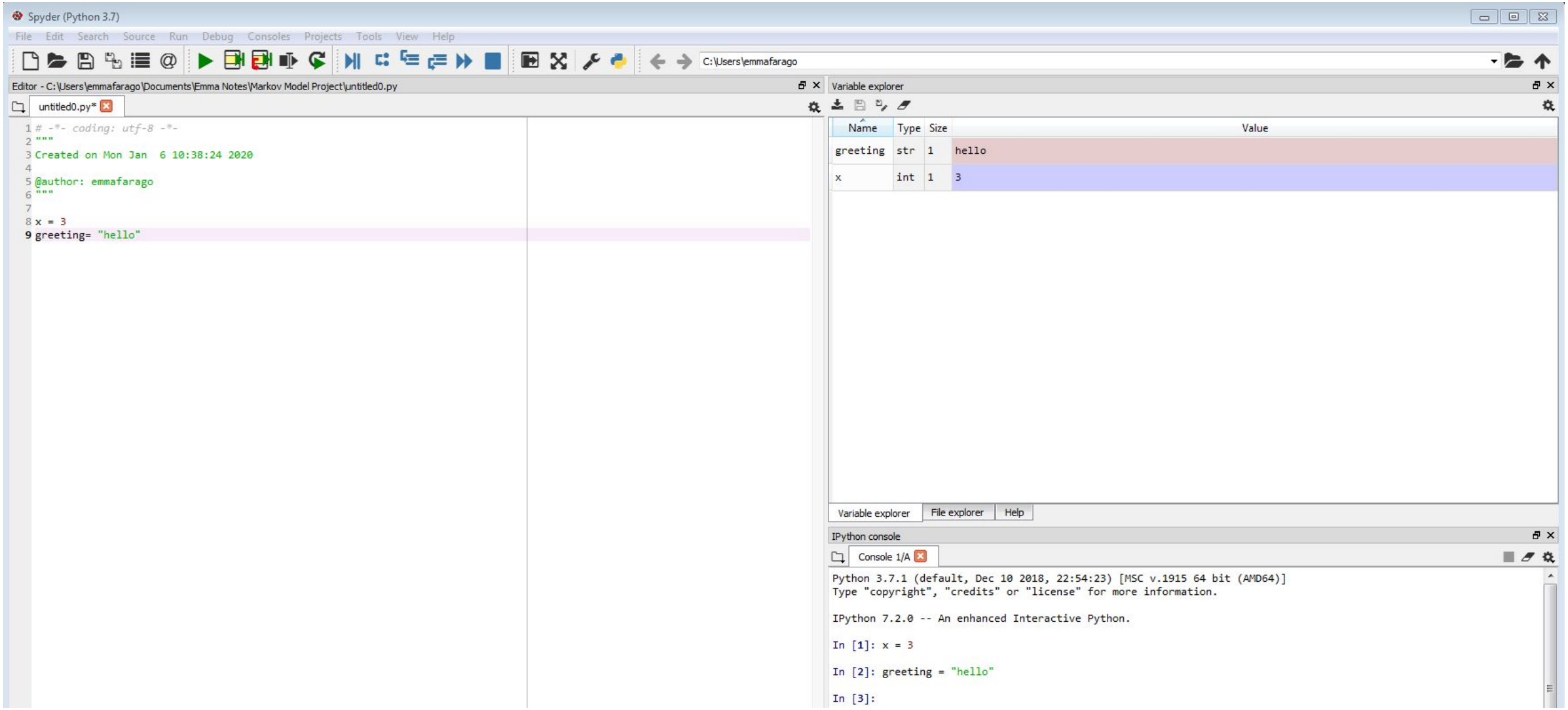
- Compiler - translates the entire program to machine code, only needs to be run once

- Interpreter - executes the program line by line as necessary, more flexible

Spyder (*recommended for this course)

- Download Anaconda (https://www.anaconda.com/distribution/)
- Conda – package and environment manager




Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\emmafarago
Editor - C:\Users\emmafarago\Documents\Emma Notes\Markov Model Project\untitled0.py
untitled0.py*
1 # -*- coding: utf-8 -*-
2 """
3 Created on Mon Jan 6 10:38:24 2020
4
5 @author: emmafarago
6 """
7
8 x = 3
9 greeting= "hello"
Variable explorer
Name Type Size Value
greeting str 1 hello
x int 1 3
Variable explorer File explorer Help
IPython console
Console 1/A
Python 3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.2.0 -- An enhanced Interactive Python.
In [1]: x = 3
In [2]: greeting = "hello"
In [3]:

Online Python Options

- Google Colab
  - https://colab.research.google.com

  - Pros:
    - Can program in your browser
    - Can share workspace online
    - Can access GPU and TPU capabilities
  - Cons:
    - Often difficult to debug/edit code

Basic Python Instructions

Variable assignments:

x = 3

greeting = “Hello world”

Printing to the screen:

print (x)

print (greeting)

print (greeting, x)

Basic Python Instructions (cont)

Operations:

x = x + 3

superhero = “bat” + “man”

User input:

newGreeting = input (“Enter a greeting”)

#need to use the “eval” function if the input is a number

newNumber = eval(input(“Enter a number”))

Python Program Example 1

- Use Python to create a program to find the area of a circle with a given radius

Input, Process, Output (IPO)

A good idea for designing simple programs is to think of the input, process, and output that you will need to build the program.

Python Program Example 1

- Use Python to create a program to find the area of a circle

Input: radius of the circle

Process (formula): A = π*r^2

Output: area of the circle

Python Program Example 2

- Use Python to create a personalized greeting for a new user by asking their name

ex: of program operating:

Computer: What is your name?
[User enters a name]
Computer: Hello [user's name]

Python Program Example 2

- Use Python to create a personalized greeting for a new user by asking their name

Input: name
Process: how to combine the username with the text “Hello”
Output: greeting

Homework

- Install a Python interpreter onto your computer and experiment with the functions we learned today: print( ) , input ( ), and with variables and operations