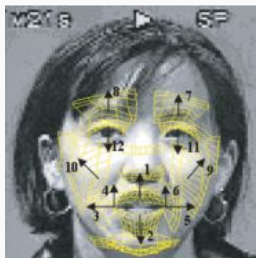


Welcome!

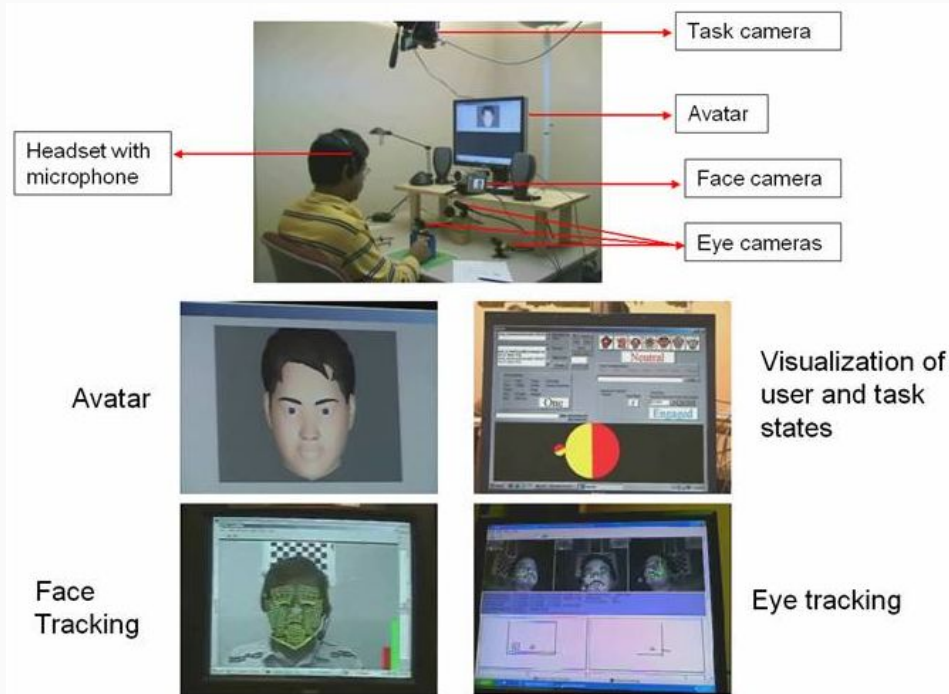
Making Game with Python (1)

Zhihong (John) Zeng & Andrew Zeng

Zhihong (John) Zeng

[illegible]

Computer Game: Human-computer Interaction



Andrew Zeng



Andrew Zeng (He/Him) · 1st

Full-Time Student at Rensselaer Polytechnic Institute

Acton, Massachusetts, United States · [Contact info](#)



**ACTON CHINESE
LANGUAGE SCHOOL**

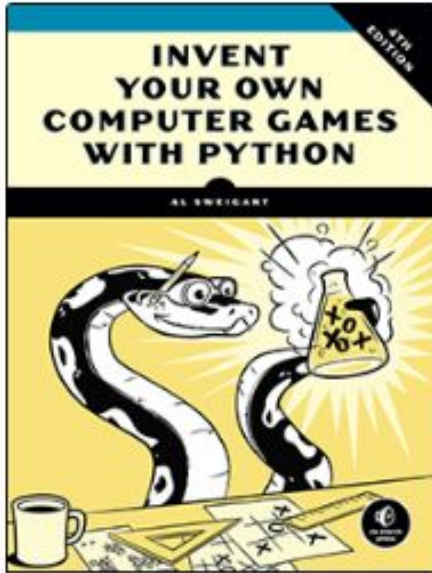


**Rensselaer Polytechnic
Institute**

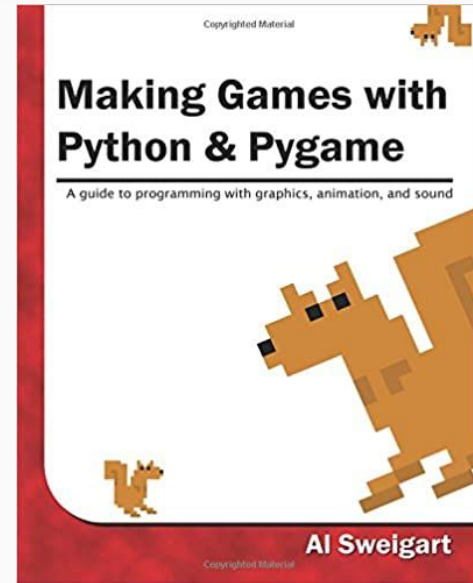
Today

- Course info
- Computer fundamentals
- Python basics
- Mathematical operations
- Python variables and types
- Exercise

Course Info

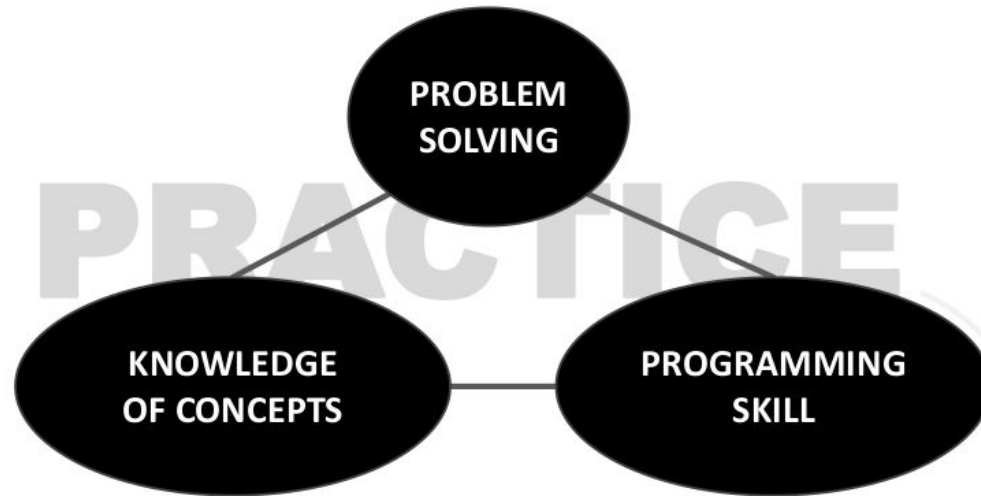


<http://inventwithpython.com/invent4thed/>



<http://inventwithpython.com/pygame/>

Course Info

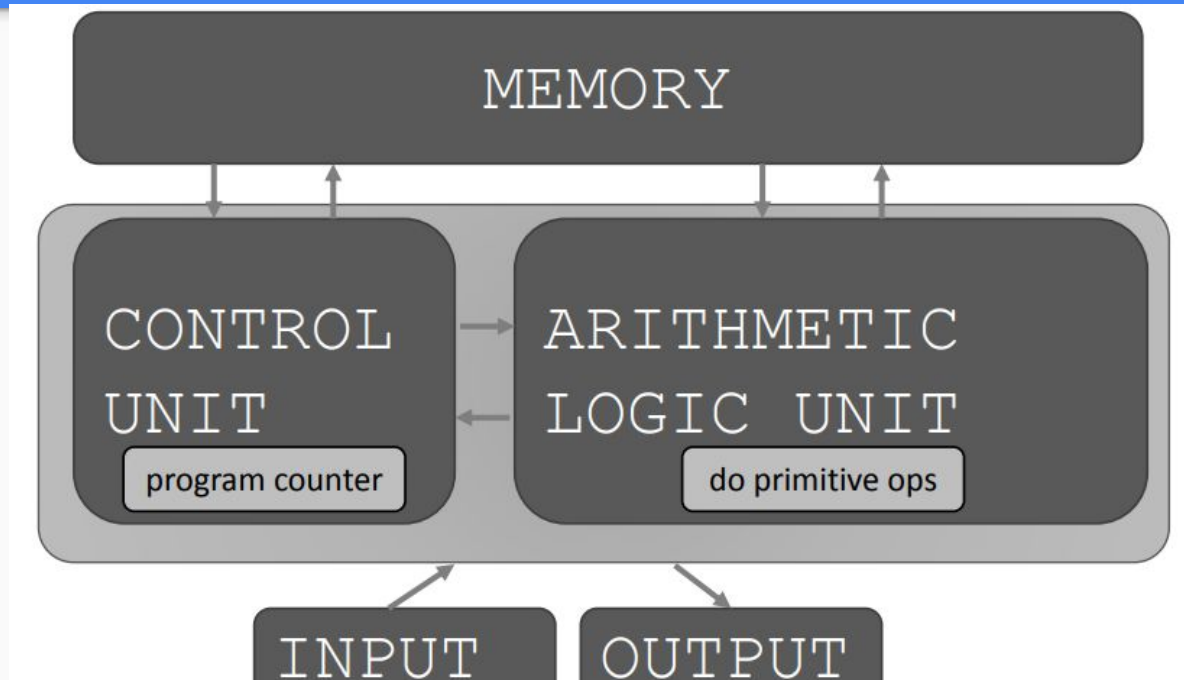


Computer Fundamentals

What does a Computer do?

- Fundamentally:
 - Performs calculations
 - Remembers results
- What kinds of calculations:
 - Built-in to the language
 - Ones that you define as the programmer
- Computers only know what you tell them

Basic Computer Architecture



What is a programming recipe

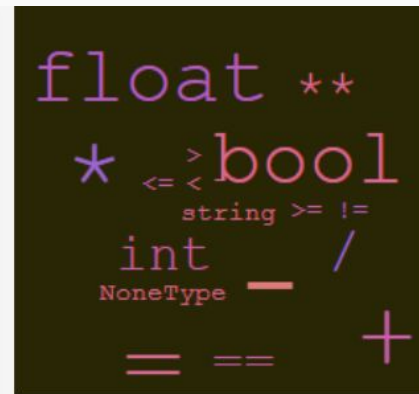
- Sequence of simple steps
- Flow of control process that specifies when each step is executed
- A means of determining when to stop

Aspects of Languages

- Primitive constructs
 - English: words
 - Programming language: numbers, strings, simple operators



Word Cloud copyright [Michael Twardos](#), All Right Reserved. This content is excluded from our Creative Commons license. For more information, see <https://ocw.mit.edu/help/faq-fair-use/>.



Word Cloud copyright unknown, All Right Reserved.
This content is excluded from our Creative Commons license. For more information, see <https://ocw.mit.edu/help/faq-fair-use/>.

Aspects of Languages

- Syntax
 - English:
 - “Cat dog boy” -> not syntactically valid
 - “Cat hugs boy” -> syntactically valid but
 - Programming language
 - “hi” 5 -> not syntactically valid
 - $3.2 * 5$ -> syntactically valid

Aspects of Languages

- Language meaning
 - English: Can have many meaning
 - “Flying planes can be dangerous”
 - Programming language: can have only one meaning but may not be what a programmer intend

Python Basics

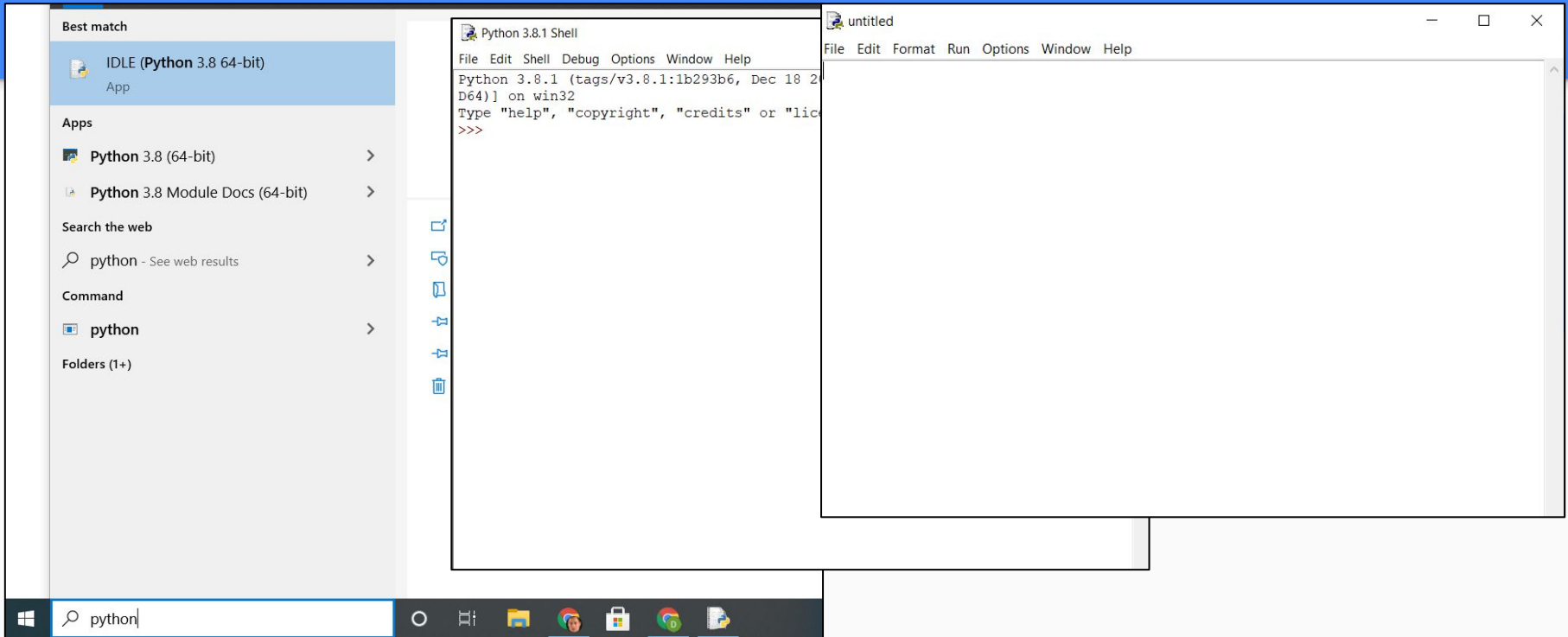
Python Programs

- A program is a sequence of definitions and commands
- A interpreter read the program to do something
- Can be typed directly in a shell or stored in a file that is read into the shell and evaluated intended

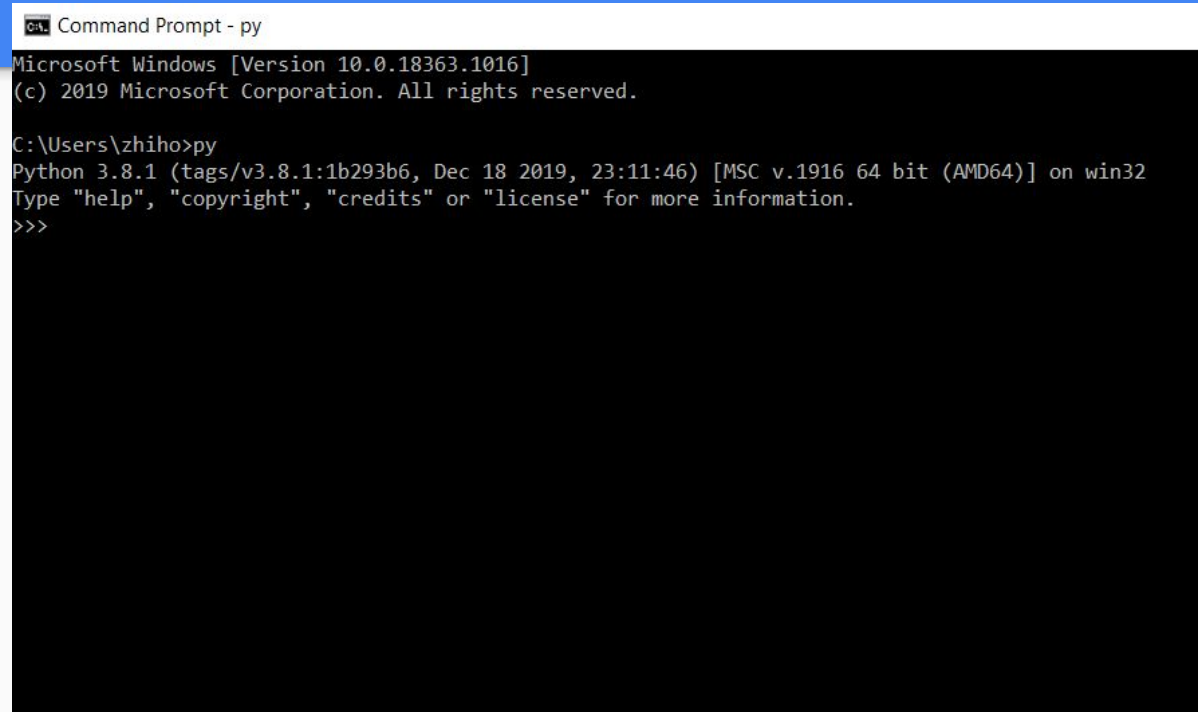
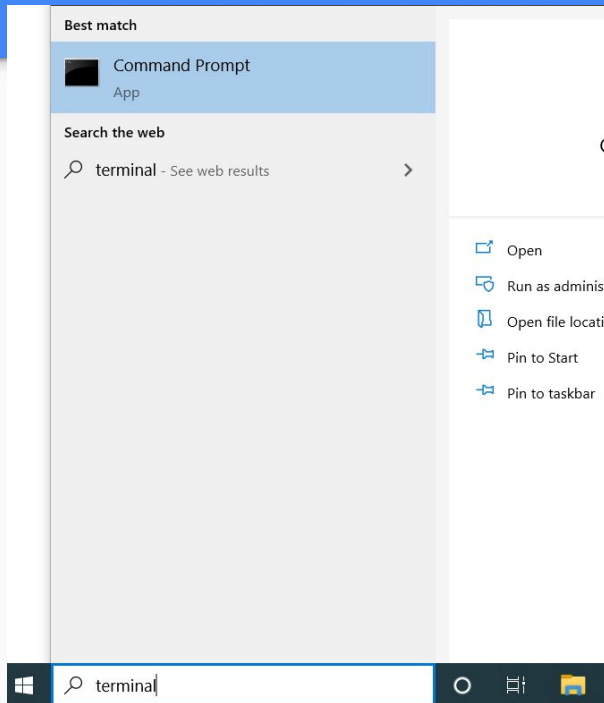
Open computer -> run python



Windows




Terminal for windows (py) and mac (python3)



Visual studio code


(<https://code.visualstudio.com/download>)

← → ↻ code.visualstudio.com/download

 This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use.


Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.




↓ Windows
Windows 7, 8, 10

User Installer	64 bit	32 bit	ARM
System Installer	64 bit	32 bit	ARM
.zip	64 bit	32 bit	ARM



↓ .deb	↓ .rpm
Debian, Ubuntu	Red Hat, Fedora, SUSE

.deb	64 bit
.rpm	64 bit
.tar.gz	64 bit



↓ Mac
macOS 10.10+

Scalar Objects

- `int` -- represent integers, ex. 5
- `float` -- represent real numbers, ex. 3.27
- `bool` -- represent Boolean values True and False
- `NoneType` -- special and has no value, None
- Can use `type()` to see the type of an object
 - `>>>type(5)` -->int
 - `>>>type(3.0)` -->float

Type Conversions (Cast)

- Can convert object of one type to another
 - `float(3)` converts integer 3 to float 3.0
 - `int(3.9)` truncates float 3.9 to integer 3
 - `int('321')` converts string '321' to integer 321
 - `str(123)` converts integer 123 to string '123'

Printing to Console

- To show output from code to a user, use print command
 - `print(3)`
 - `print(3+2)`
 - `print('ABC')`

Expressions

- Combine objects and operators to form expressions
- Syntax for a simple expression
 - `<object> <operator> <object>`

Operators on ints and floats

- $i + j$ → addition (e.g., $1+2$)
- $i - j$ → subtraction (e.g., $2-1$)
- $i * j$ → product (e.g., $2*4$)

Note: if both are ints, result is int. If either or both are floats, result is float

- i / j → division (e.g., $4/2$).
- $i \% j$ → remainder when i is divided by j (e.g., $5\%2$)
- $i ** j$ → i to the power of j (e.g., $2**3$)

Operation precedence (order)

- Parentheses used to tell Python to do these operations first
- Operator precedence without parentheses
 - `**`
 - `*`
 - `/`
 - `+` and `-` - executed left to right, as appear in expression

Binding Variables and Values

- Equal sign is an assignment of a value to a variable name
 - `pi = 3.14`
 - `text = 'abc' or "abc"`
- Value stored in computer memory
- An assignment binds name to value
- Retrieve value associated with name or variable by invoking the name
 - `print(pi)`
 - `print(text)`

Variables are very useful

- Why give names to values of expressions
 - To reuse names instead of values
 - Easier to change code later

```
pi = 3.14  
radius = 2.2  
area = pi * (radius ** 2)
```

Programming vs Math

- In programming, you do not “solve for x”
- Programming:
 - Expression on the right
 - Variable name on the left

Python:

```
pi = 3.14  
radius = 2.2  
area = pi * (radius ** 2)
```

Math:

```
pi = 3.14  
radius = 2.2  
pi * (radius ** 2) = area
```