[301] Programming

Tyler Caraza-Harter

Learning Objectives

Skills:

- Run Python
- Run PyCharm

Learn common Python operators:

- Mathematical (e.g., “+” and “-“)
- Comparison (e.g., “==” and “>”)
- Logical (e.g., “and” and “not”)

Learn about different data types:

- int, float, str, bool

Learn about boolean logic

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos

What you need to write/run code

An interpreter

- Python 3 (not Python 2)
- We prefer you install Python 3 with Anaconda
(Anaconda is not strictly necessary yet)

An editor

- Which one doesn’t matter much
- PyCharm is a good choice, and is installed in the labs

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)

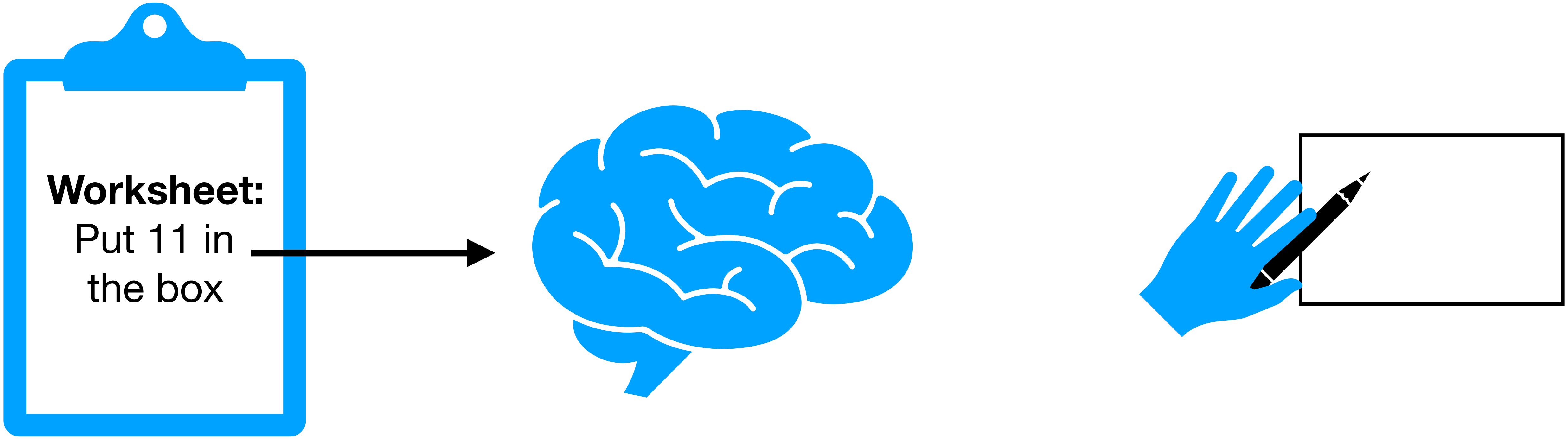
Worksheet:
Put 11 in
the box

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



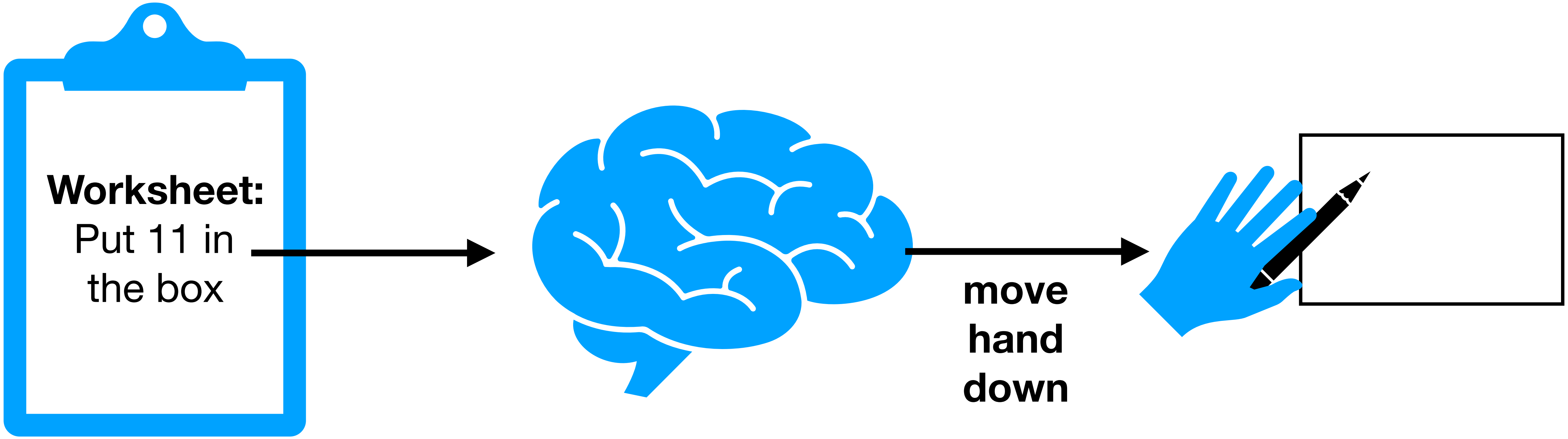
Worksheet:
Put 11 in the box

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in
the box

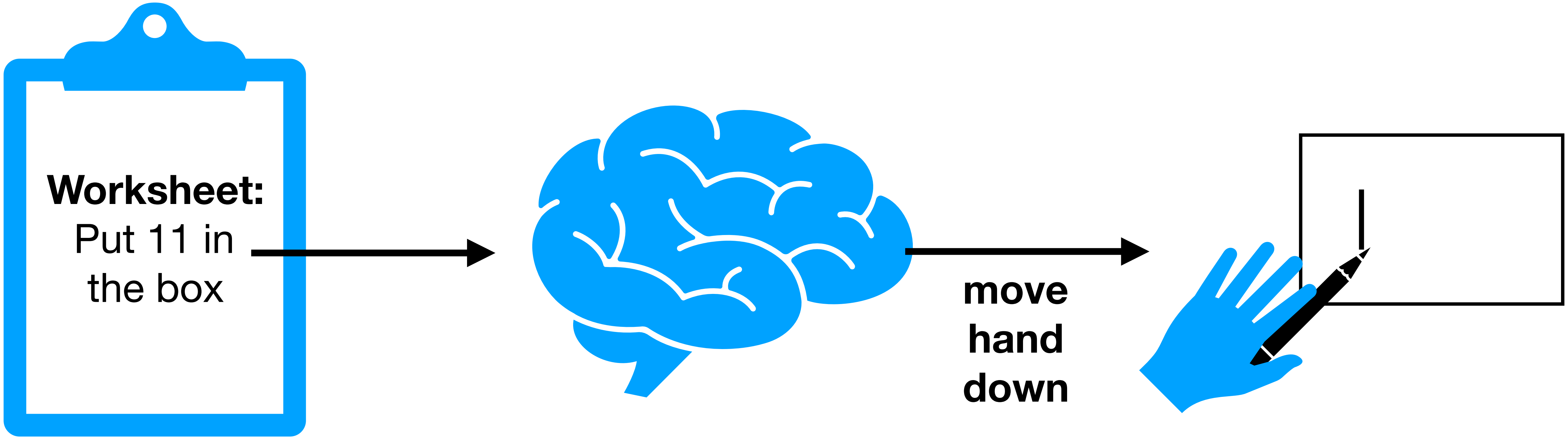
**move
hand
down**

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in
the box

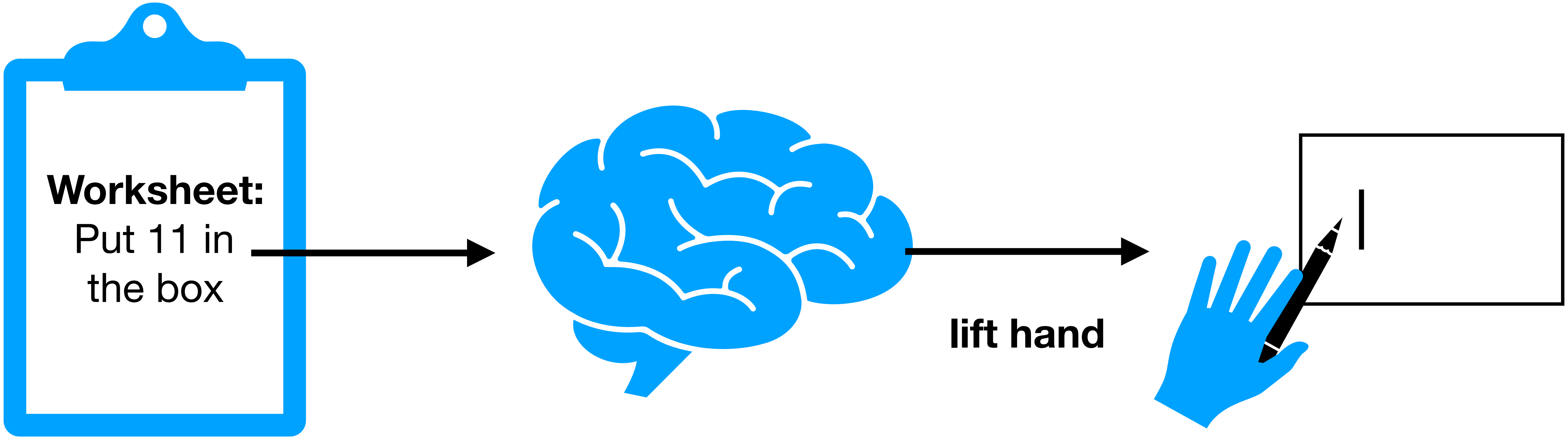
**move
hand
down**

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet: Put 11 in the box

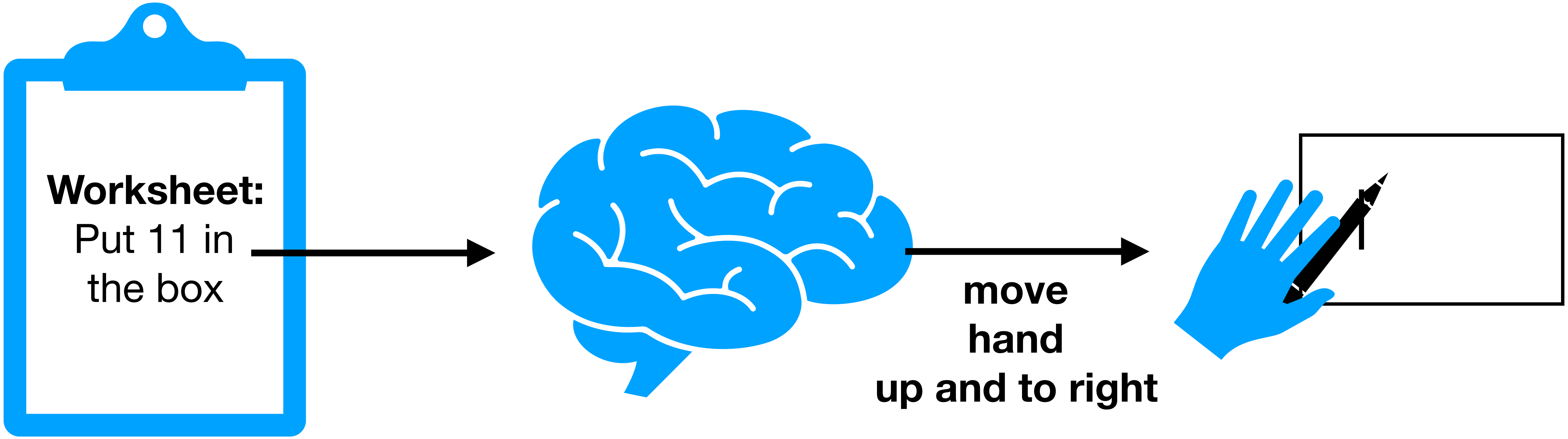
lift hand

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in
the box

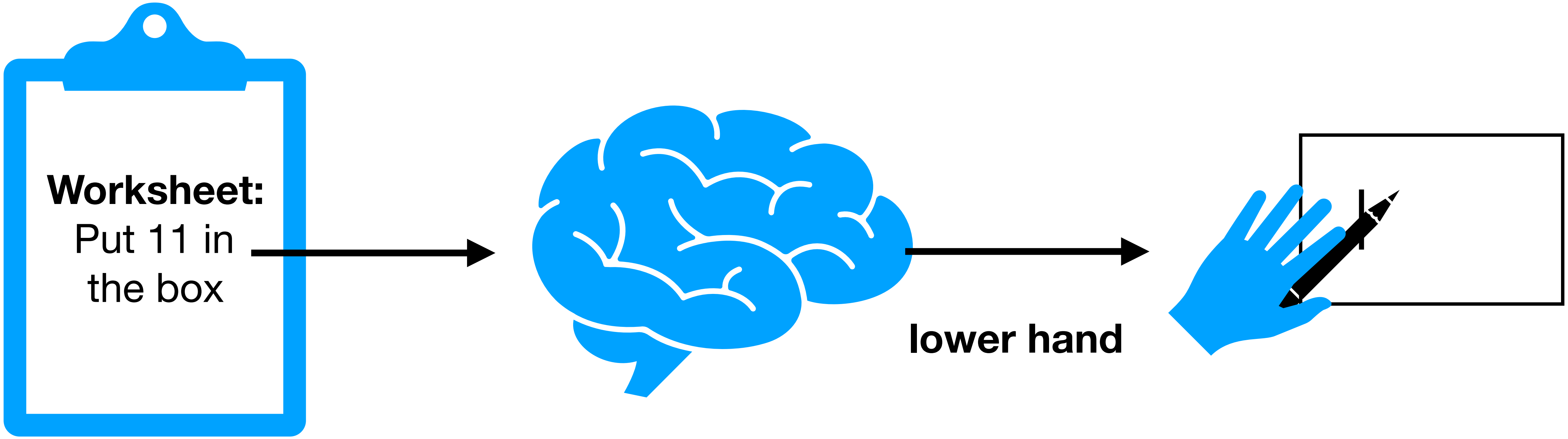
**move
hand
up and to right**

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet: Put 11 in the box

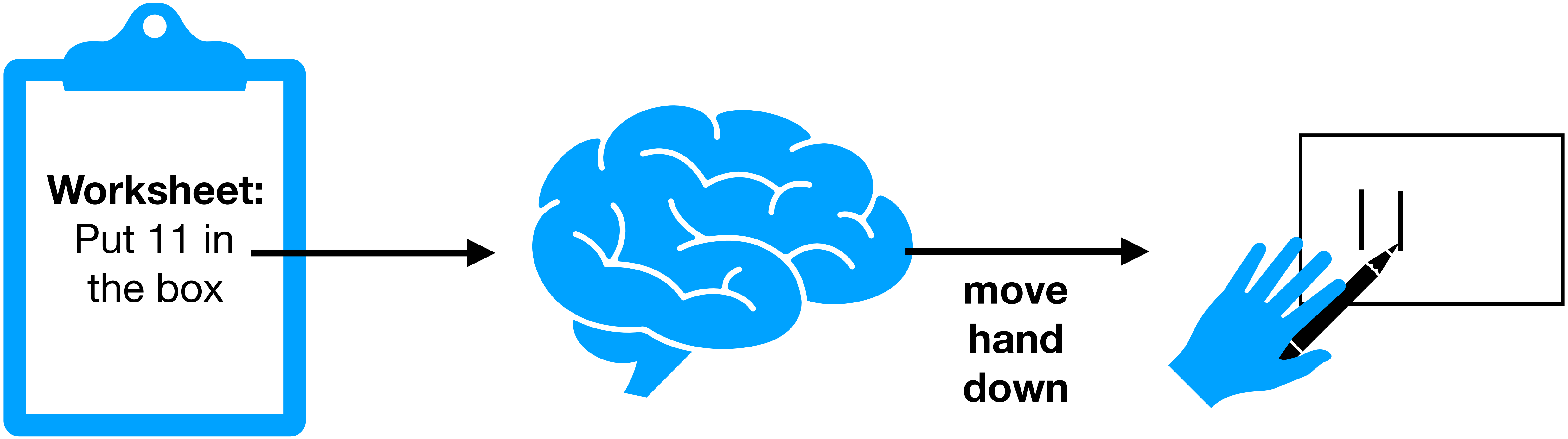
lower hand

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in
the box

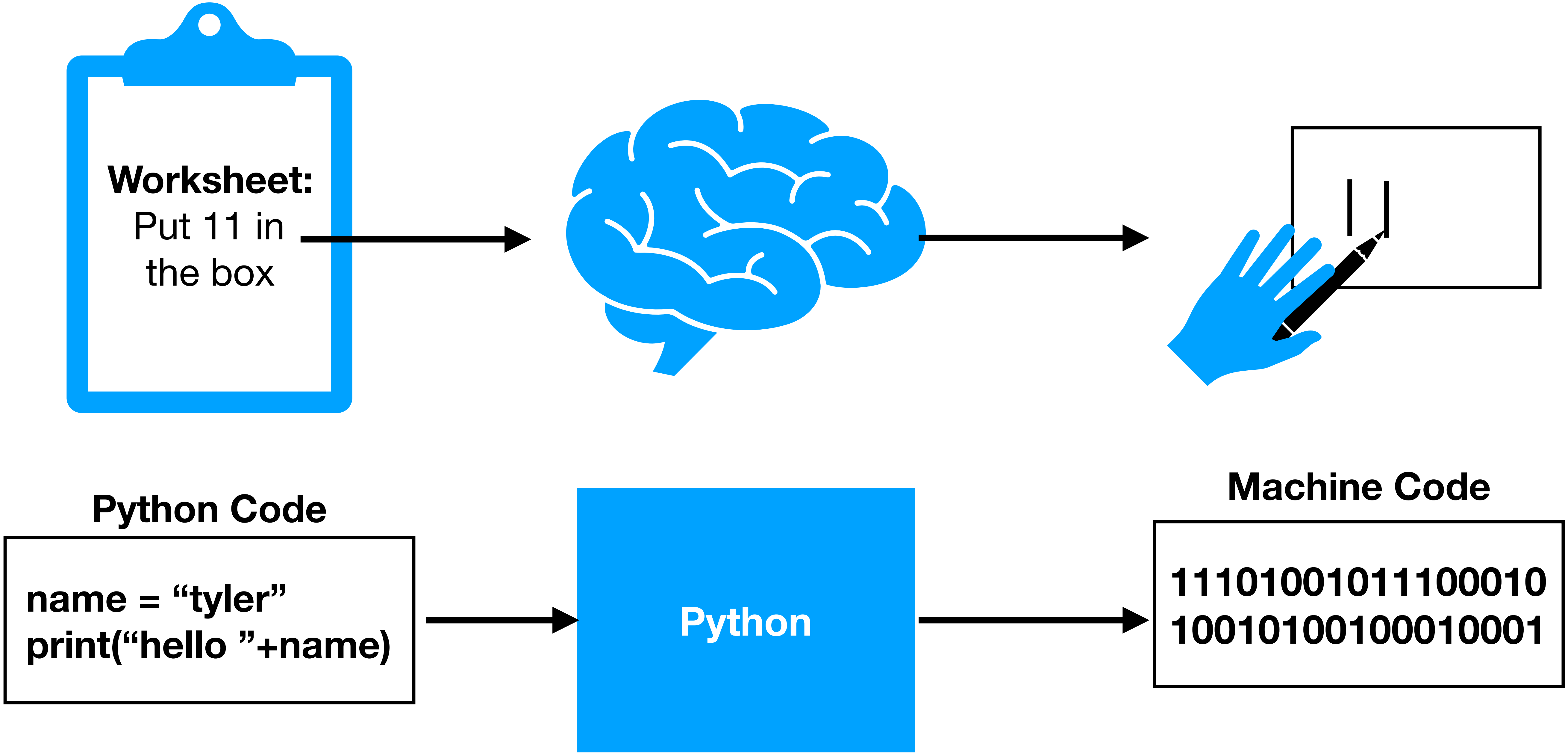
move
hand
down

You were an interpreter when you did the pseudocode worksheets!

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in the box

Python Code

```
name = “tyler”
print(“hello ”+name)
```

Python

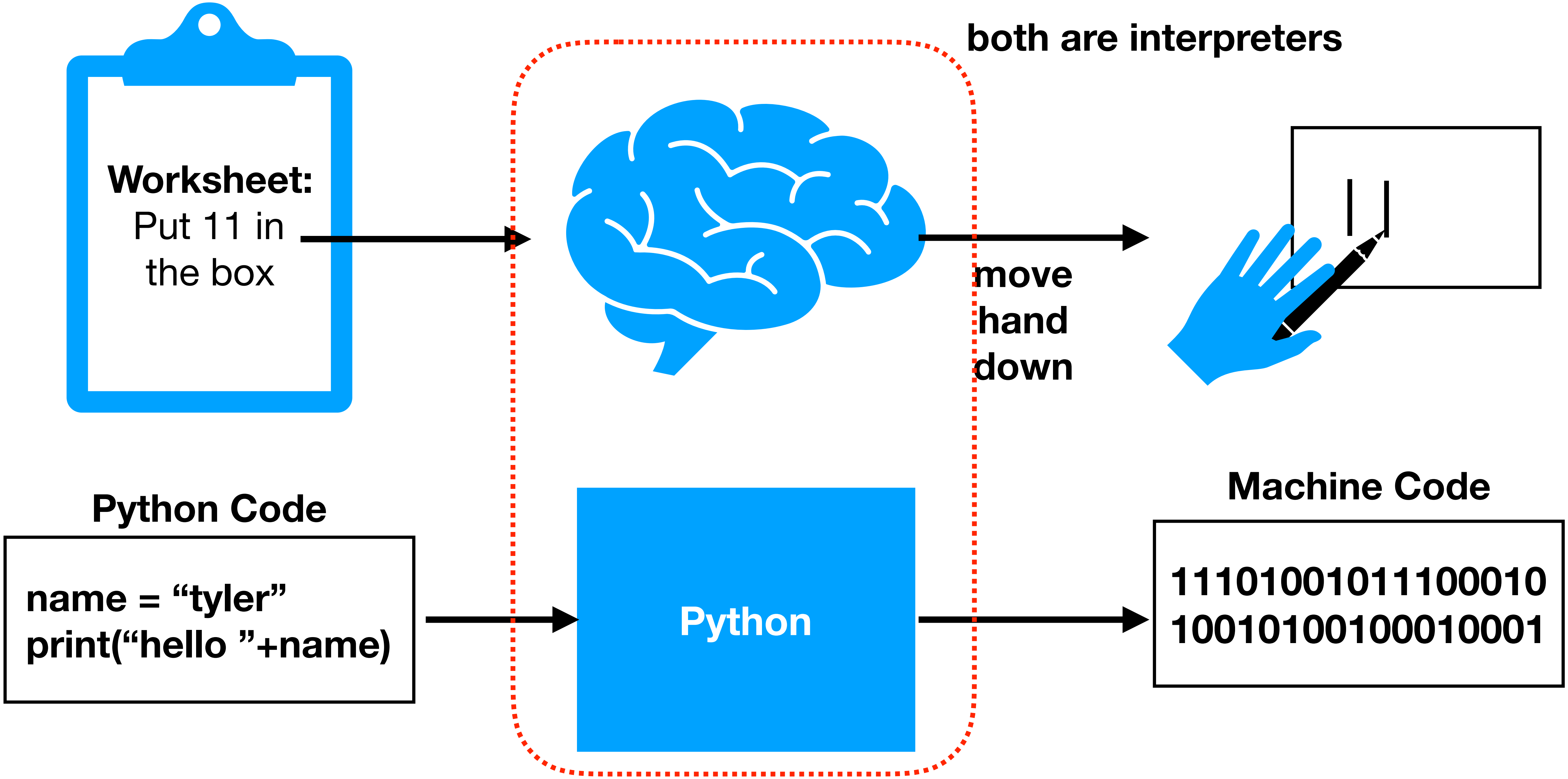
Machine Code

```
11101001011100010
10010100100010001
```

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



Worksheet:
Put 11 in the box

both are interpreters

move hand down

Python Code

```
name = “tyler”
print(“hello ”+name)
```

Python

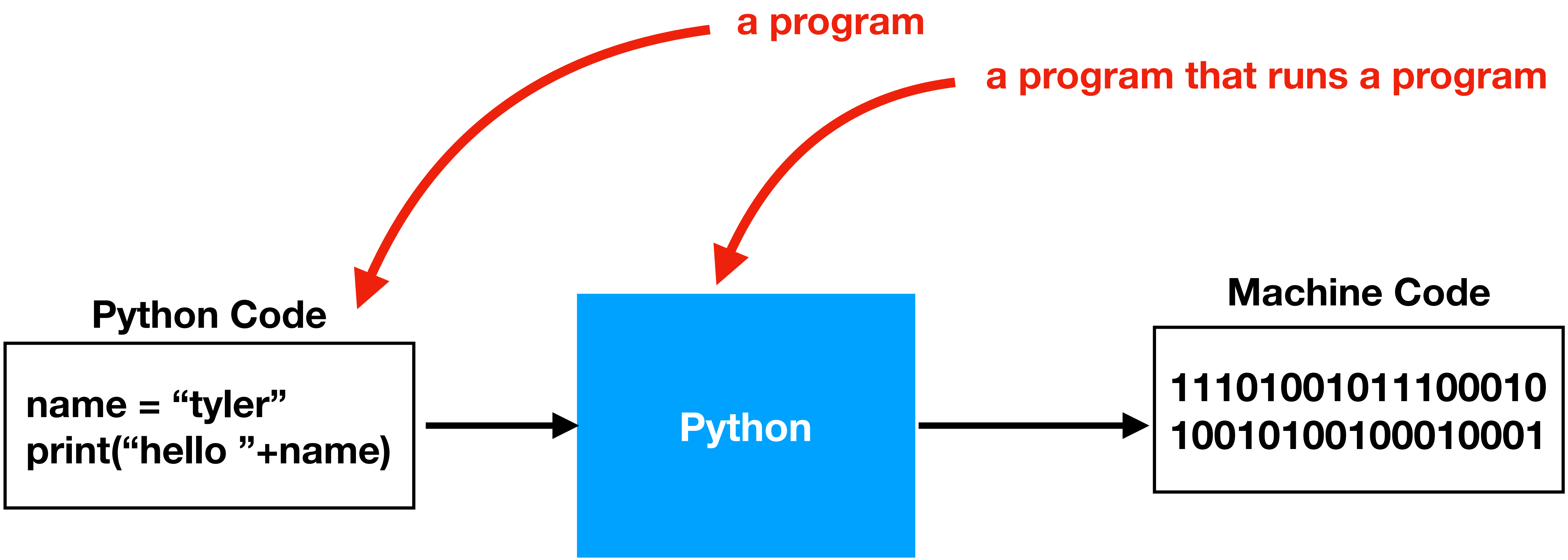
Machine Code

```
11101001011100010
10010100100010001
```

Interpreter

A program that runs a program

- Translates something human likes (nice Python code) to something the machine likes (ONEs and ZEROs)



a program

a program that runs a program

Python Code

```
name = “tyler”
print(“hello ”+name)
```

Python

Machine Code

```
11101001011100010
10010100100010001
```

Editor

Program for writing code and other simple files

- Many different editors could be used to write the same code, just like many different web browsers could access the same site

Editor

Program for writing code and other simple files

- Many different editors could be used to write the same code, just like many different web browsers could access the same site
- Why does it matter what you use?

Editor

Program for writing code and other simple files

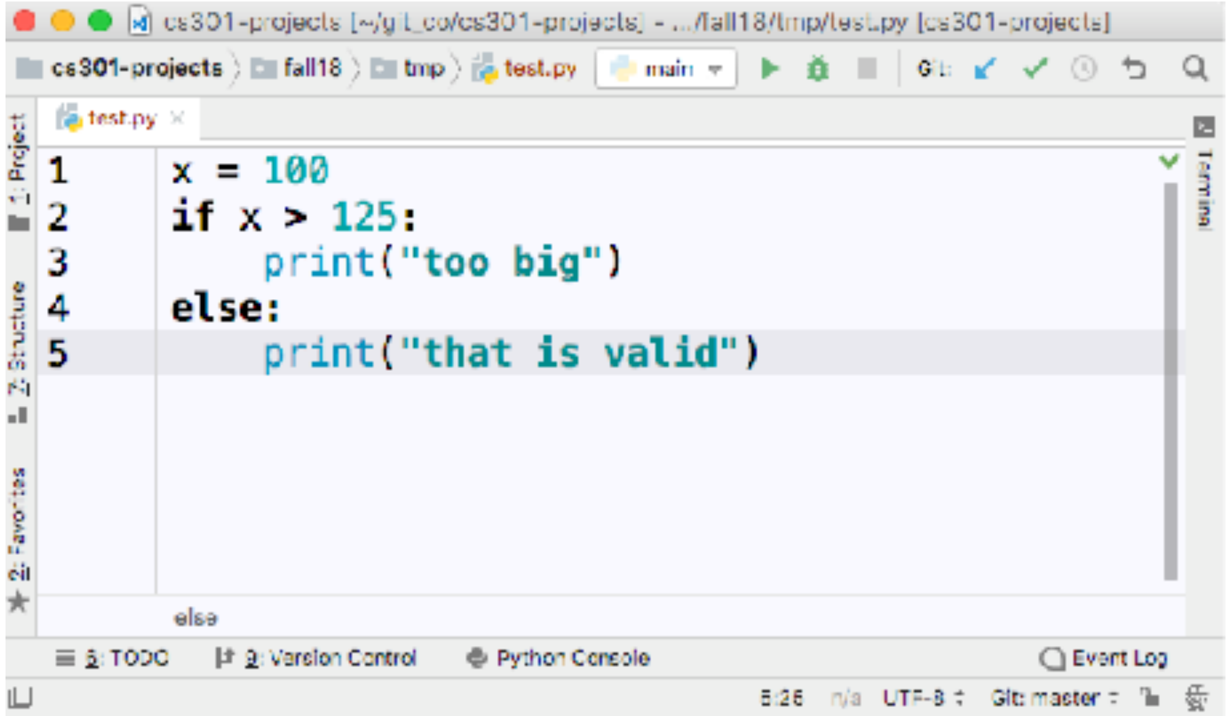
- Many different editors could be used to write the same code, just like many different web browsers could access the same site
- Why does it matter what you use?
    1. Some have a builtin terminal

Editor

Program for writing code and other simple files

- Many different editors could be used to write the same code, just like many different web browsers could access the same site
- Why does it matter what you use?
    1. Some have a builtin terminal
    2. They add helpful color to your code

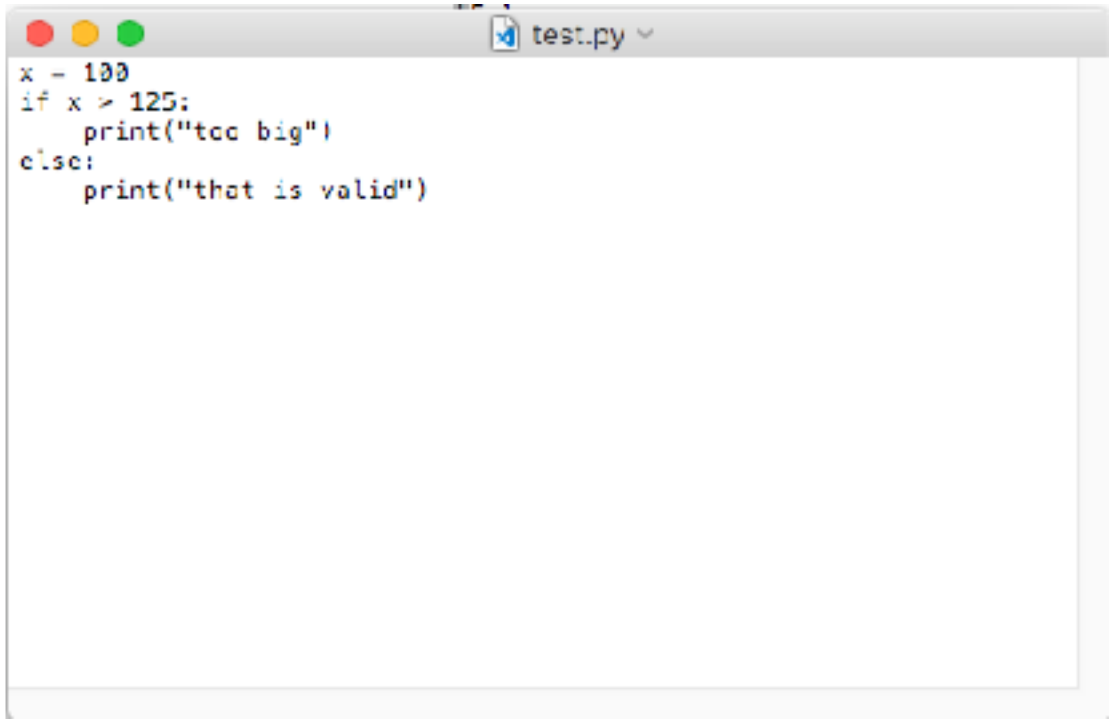
PyCharm



```
x = 100
if x > 125:
    print("too big")
else:
    print("that is valid")
```

else

TextEdit



```
x = 100
if x > 125:
    print("too big")
else:
    print("that is valid")
```

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** ~~(1/2)~~

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** ~~(1/2)~~

3 * 3 + 2 * 2 + 16 ** (**0.5**)

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)
3 * 3 + 2 * 2 + ~~16 ** (0.5)~~

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + ~~16 ** (0.5)~~

3 * 3 + 2 * 2 + **4**

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)
3 * 3 + 2 * 2 + 16 ** (0.5)
~~3 * 3~~ + 2 * 2 + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

~~3 * 3~~ + 2 * 2 + 4

9 + 2 * 2 + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

3 * 3 + 2 * 2 + 4

9 + ~~2 * 2~~ + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

3 * 3 + 2 * 2 + 4

9 + ~~2 * 2~~ + 4

9 + **4** + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

3 * 3 + 2 * 2 + 4

9 + 2 * 2 + 4

~~9 + 4~~ + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

3 * 3 + 2 * 2 + 4

9 + 2 * 2 + 4

~~9 + 4~~ + 4

13 + 4

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)
3 * 3 + 2 * 2 + 16 ** (0.5)
3 * 3 + 2 * 2 + 4
9 + 2 * 2 + 4
9 + 4 + 4
~~13 + 4~~

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Order of Simplification

Python works by simplifying, applying one operator at a time

3 * 3 + 2 * 2 + 16 ** (1/2)

3 * 3 + 2 * 2 + 16 ** (0.5)

3 * 3 + 2 * 2 + 4

9 + 2 * 2 + 4

9 + 4 + 4

~~13 + 4~~

17

Rules

- First work within parentheses
- Do higher precedence first
- Break ties left to right

Operator Precendence

What is it?	Python Operator
exponents	**
signs	+x, -x
multiply/divide	*, /, //, %
add/subtract	+, -
comparison	==, !=, <, <=, >, >=
boolean stuff	not
…	and
…	or

simplify first

simplify last

these are the ones you should be learning at this point in the semester (there are a few more not covered now)

Operator Precendence

	What is it?	Python Operator	
Mathematical	exponents	**	simplify first
	signs	+x, -x	
	multiply/divide	*, /, //, %	
	add/subtract	+, -	
	comparison	==, !=, <, <=, >, >=	
Logic	boolean stuff	not	
	…	and	
	…	or	simplify last

these are the ones you should be learning at this point in the semester (there are a few more not covered now)

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos

Today's Outline

Software

- Interpreters
- Editors

Demos

Operator Precedence

Demos

Boolean Logic

Demos