Lecture 1: Introduction

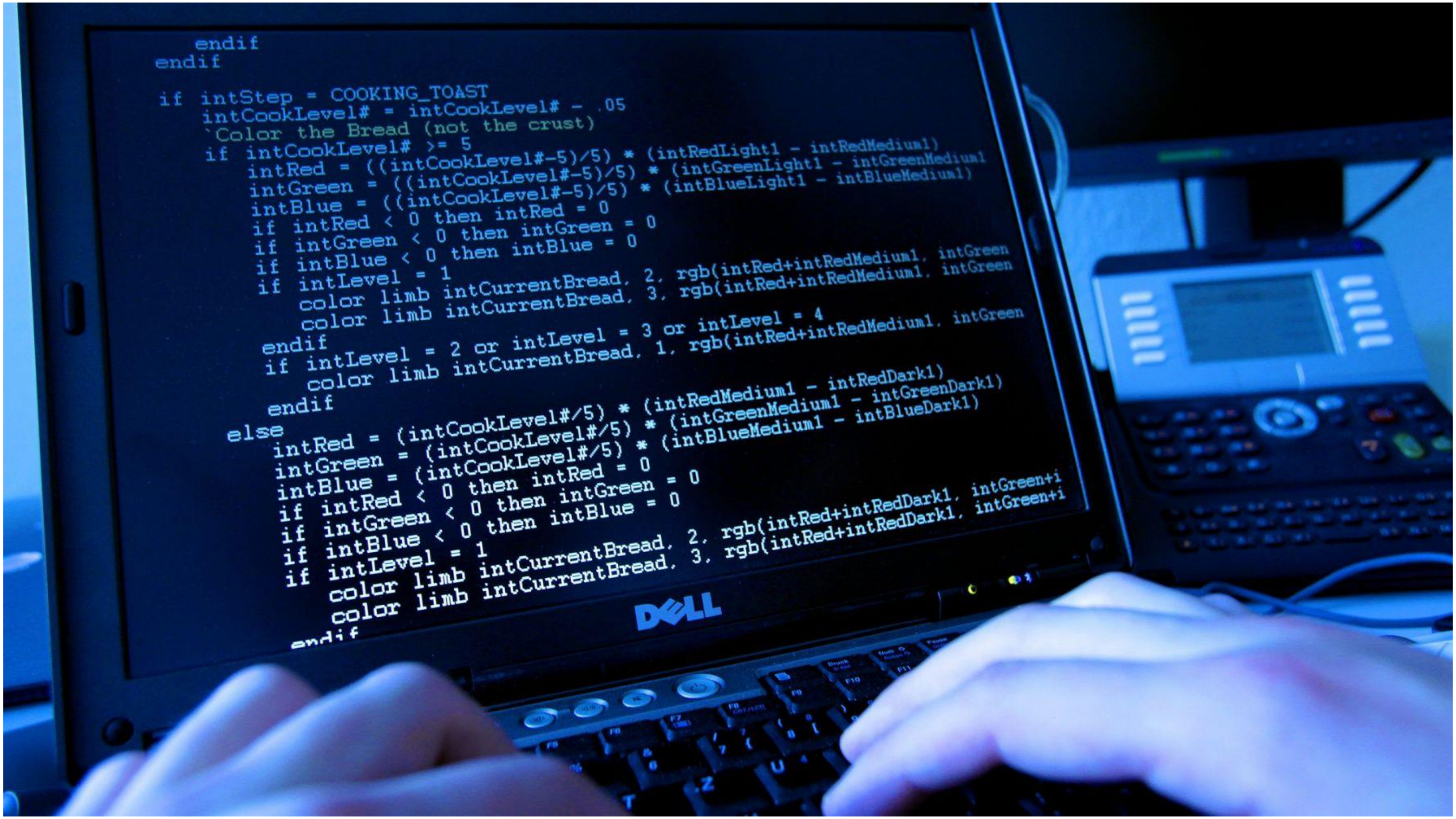
CS 50

January 20, 2026

Computer Science

CS != Programming

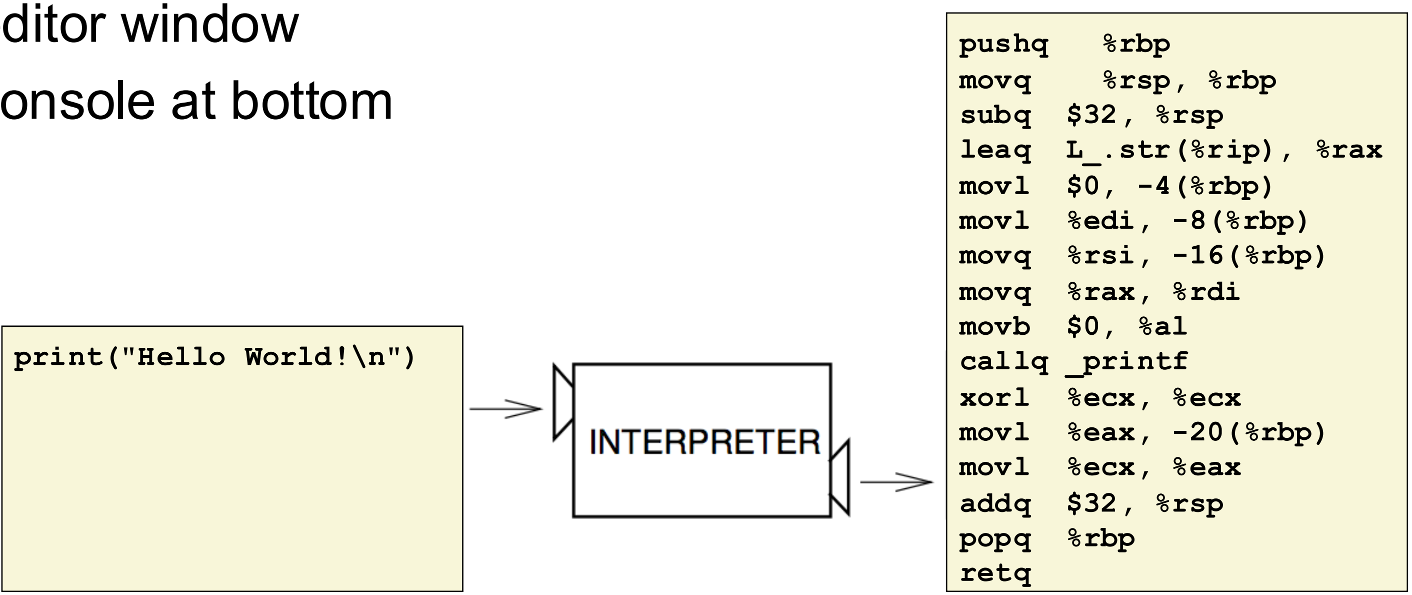
Programming



```
        endif
    endif

    if intStep = COOKING_TOAST
        intCookLevel# = intCookLevel# - .05
        `Color the Bread (not the crust)
        if intCookLevel# >= 5
            intRed = ((intCookLevel#-5)/5) * (intRedLight1 - intRedMedium1)
            intGreen = ((intCookLevel#-5)/5) * (intGreenLight1 - intGreenMedium1
            intBlue = ((intCookLevel#-5)/5) * (intBlueLight1 - intBlueMedium1)
            if intRed < 0 then intRed = 0
            if intGreen < 0 then intGreen = 0
            if intBlue < 0 then intBlue = 0
            if intLevel = 1
                color limb intCurrentBread, 2, rgb(intRed+intRedMedium1, intGreen
                color limb intCurrentBread, 3, rgb(intRed+intRedMedium1, intGreen
            endif
            if intLevel = 2 or intLevel = 3 or intLevel = 4
                color limb intCurrentBread, 1, rgb(intRed+intRedMedium1, intGreen
            endif
        else
            intRed = (intCookLevel#/5) * (intRedMedium1 - intRedDark1)
            intGreen = (intCookLevel#/5) * (intGreenMedium1 - intGreenDark1)
            intBlue = (intCookLevel#/5) * (intBlueMedium1 - intBlueDark1)
            if intRed < 0 then intRed = 0
            if intGreen < 0 then intGreen = 0
            if intBlue < 0 then intBlue = 0
            if intLevel = 1
                color limb intCurrentBread, 2, rgb(intRed+intRedDark1, intGreen+i
                color limb intCurrentBread, 3, rgb(intRed+intRedDark1, intGreen+i
            endif
```

Python

- Python 3.14
- Visual Studio
  - projects
  - editor window
  - console at bottom



```
print("Hello World!\n")
```

```
pushq   %rbp
movq    %rsp, %rbp
subq  $32, %rsp
leaq  L_.str(%rip), %rax
movl  $0, -4(%rbp)
movl  %edi, -8(%rbp)
movq  %rsi, -16(%rbp)
movq  %rax, %rdi
movb  $0, %al
callq _printf
xorl  %ecx, %ecx
movl  %eax, -20(%rbp)
movl  %ecx, %eax
addq  $32, %rsp
popq  %rbp
retq
```

Types

A **type** is a set of possible values

integer (`int`)

- Values: 0, 1, -10, 34022, …

string (`str`)

- Values: "Hi!", "", "2.0",…

All values have types.
Integers and strings are two common types

You can determine the type of a value using the command `type(<value>)`

Types

A **type** is a set of possible values and a set of operations that can be performed on those values

integer (`int`)

- Values: 0, 1, -10, 34022, …
- Operations: +, -, /, *
  ** (exponent),
  % (remainder)
  // (truncated division)

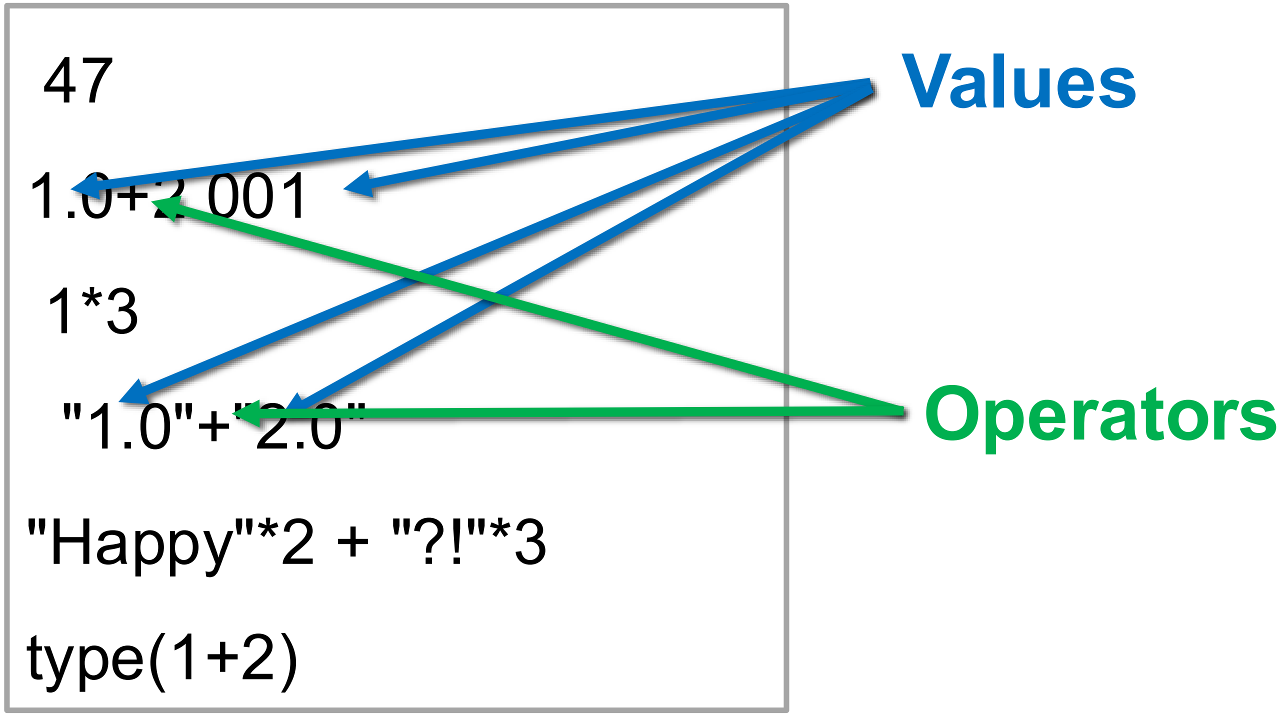
string (`str`)

- Values: "Hi!", "", "2.0",…
- Operations: + (concatenation)
  * (duplication),

All values have types
Common types: int, float, str, bool

Expressions

- Expressions represent a value
- Python evaluates expressions (similar to a calculator)



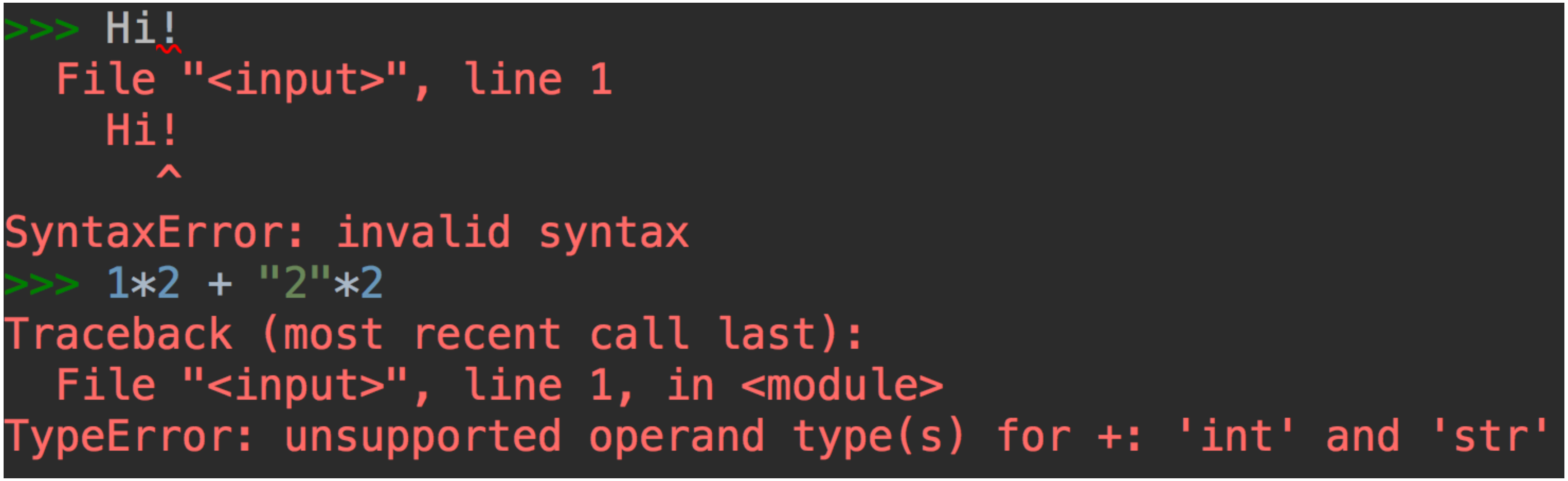
```
47
1.0+2.001
1*3
"1.0"+"2.0"
"Happy"*2 + "?!"*3
type(1+2)
```

Exercise 1: Expressions

```
1/2

13

4 + 3 * 2

( "A"*2 + "?"*3 ) * 2

14 % 5

5 ** 2

Hi!

1*2 + "2"*2
```

Errors

- Three types of errors:
  - ValueError: unsupported operand value(s)
  - TypeError: unsupported operand type(s) for +: 'int' and 'str'
  - SyntaxError: invalid syntax



```
>>> Hi!
  File "<input>", line 1
    Hi!
      ^
SyntaxError: invalid syntax
>>> 1*2 + "2"*2
Traceback (most recent call last):
  File "<input>", line 1, in <module>
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Exercise 2: Expressions and Errors

```
3 * "5"

str(3) * int("5")

1 / 2 * "Hello"

 "1.0" + 2.0

 int("2") * "2"

 1 ** 2.5

str("2") * 4
```

Types and Functions

- How to find out the type?

```
"Happy"*2 + "?!"*3

1*2 + "2"*2
```

- What if you want to change the type?

- Type functions
  - type( )
  - str( ), int( ), float( )

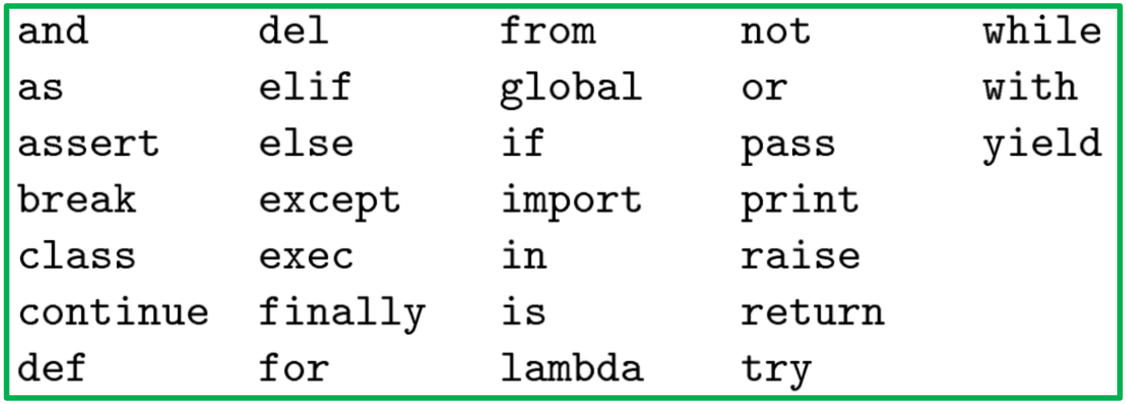
```
type(3.0)

1*2 + int("2")*2

str(1)*2 + "2"*2
```

Example: Word Problem

If you run a 10 kilometer race in 43 minutes 30 seconds, what is your average time per mile? (Hint: there are 1.61 kilometers in a mile).

Variables

- A variable is a name that refers to a value
  - names should be meaningful
  - by convention words separated by an underscore
  - names cannot be a keyword (e.g. *return*), cannot include spaces, must begin with a letter



and
as
assert
break
class
continue
def
del
elif
else
except
exec
finally
for
from
global
if
import
in
is
lambda
not
or
pass
print
raise
return
try
while
with
yield

Assigning variables

- Can assign a value to a variable
- Right hand side can be any expression (anything that is, or that evaluates to, a value)

```
x = 13

a_string = "Hello”

y = (47 + 13) / 2

another_string = str(20) + str(26)
```

Variables and Expressions

- a variable evaluates to the value stored in that variable
- variables can be used in expressions

```
x = 13

y = (47 + x) / 2
```

Example: Writing a Program

If you run a 10 kilometer race in 43 minutes 30 seconds, what is your average time per mile? (Hint: there are 1.61 kilometers in a mile).

Exercise 3: Writing a Program

If you run a 10 kilometer race in 43 minutes 30 seconds, what is your average speed in miles per hour? (Hint: there are 1.61 kilometers in a mile).

Demo: Running a Program

Exercise 4: More Programming

You're having a party and you need to figure out how much food to buy. Here's what you know:

- *Teran isn't a big fan of hot dogs, so he'll only eat 1*
- *Jasmin generally eats 2*
- *Chris always eats twice as many as Jasmin*
- *Brenda always eats one less than Chris*
- *Grace eats half as many as Brenda at the party but also likes to bring one home*

Write a program to calculate how many hot dogs you need.

Course Logistics

Prof. Eleanor Birrell

Prof. Dave Kauchak

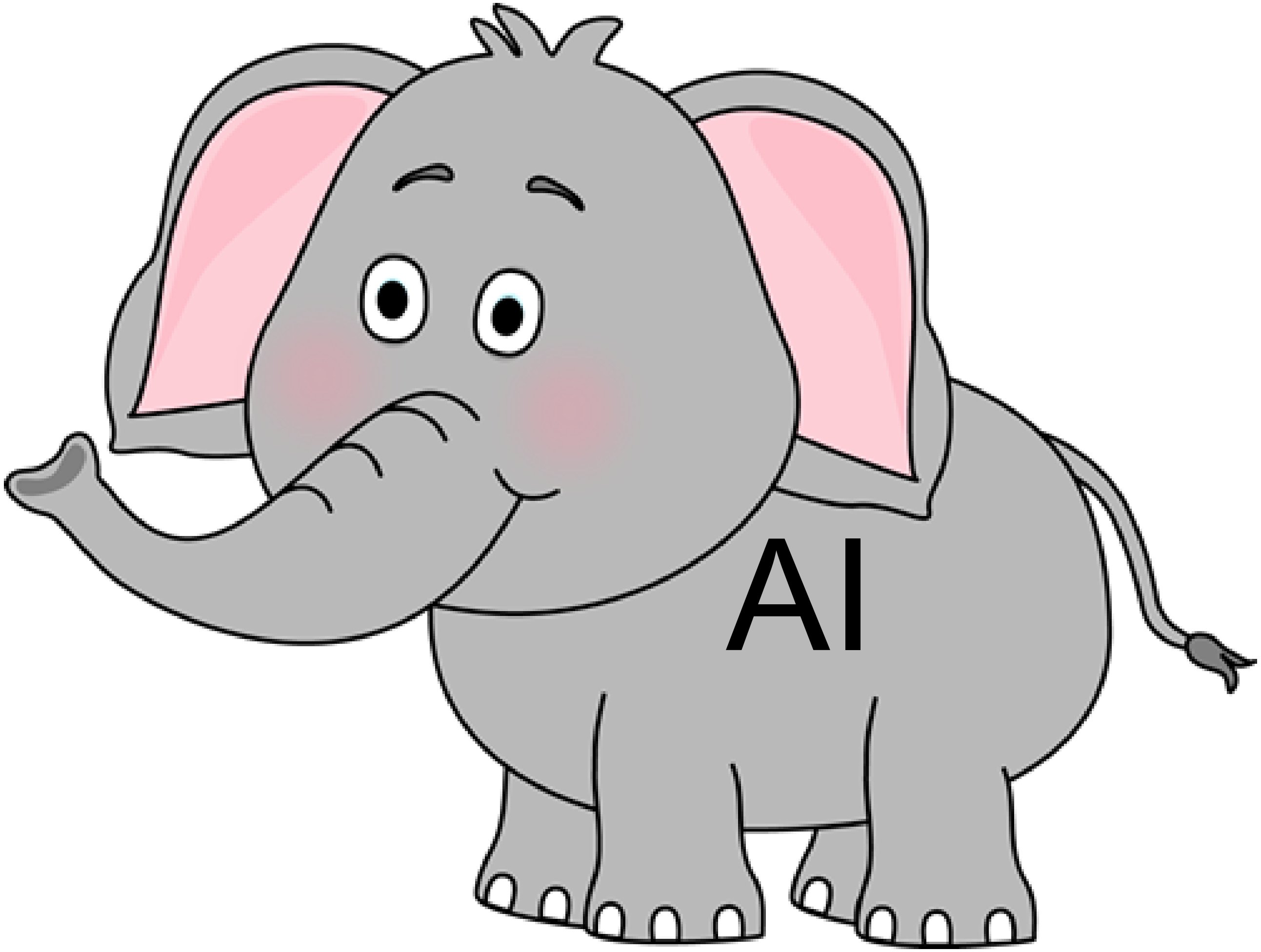
- **Class Meetings (Section 2):**
  - Tuesdays and Thursdays, 11am-12:15pm in Edmunds 114

Course Work

- Homework Assignments (30%)
  - Approximately 6 assignments
  - Designed to help you learn the material
  - Due on Mondays at 11:59pm PT

- Midterm (30%)
  - Thursday, February 19 (in class)

- Final (40%)
  - Thursday, March 12 (in class)
  - Cumulative
  - Show us what you’ve learned!


AI

Course Website

All information is on the Course website:
https://cs.pomona.edu/classes/cs50/section2/

50 communications, community, q&a, etc. on slack:
#cs50-sec2-2026sp

CS Community

- Slack (Everyone!) pomonacs.slack.com
- BBICS
- WACM