|  |
| --- |
| 15-110 Recitation 03 |

# **Recap**

* Tokenizing/parsing/translating
* Bytecode
* Return/print/side effects
* if/elif/else statements, nested if statements, functions

and calling functions circuits and gates

* Error types
* Circuits and gates

# **Reminders for Students**

* Check2 due Monday 9/21 @ Noon EDT!
* HW 1 grades published! Resubmissions due 9/22 @ Noon EDT
* First Test is Wednesday 9/23
* Check grades on Gradescope/canvas

Post-Recitation Feedback Form: <https://forms.gle/NmBJdt4zwTxgsjuw8>

|  |
| --- |
| Problems |

# **BYTECODE REVIEW**

How does the interpreter convert Python code into machine executable instructions?

Answer/Notes:

The program below is translated into the following set of bytecode instructions. Run through the bytecode using a value stack and fill in the table on the right (Refer to Week 2-3 slides).

x = 7

y = x - 5

z = y

LOAD\_CONST 0 *# 7*

STORE\_NAME 0 *# x*

LOAD\_NAME 0

LOAD\_CONST 1 *# 5*

BINARY\_SUBTRACT

STORE\_NAME 1 *# y*

LOAD\_NAME 1

STORE\_NAME 2 *# z*

|  |
| --- |
| Variable Table |
| id | **name** | **value** |
| 0 |  |  |
| 1 |  |  |
| 2 |  |  |

|  |
| --- |
| Literal Table |
| id | **value** |
| 0 |  |
| 1 |  |

|  |
| --- |
|  |
|  |
|  |
| **Value Stack** |

# **PRINT VERSUS RETURN**

Let’s review the following lines of code:

num1 = 8

num2 = 4

def multiply (a, b):

 a\*b

multiply(num1, num2)

What will these lines output? →

Notes:

# **ERRORS**

Identify the type of error and its category:

 1/0 → Error:

 def my\_new\_function(argument1, argument2,) → Error:

 '2' + 2 → Error:

 print("hello world) → Error:

 a = 3 + 5 7 → Error:

 def average(num1, num2, num3): → Error:

 return num1 + num2 + num3 / 3

# **CONTROL FLOW TO CODE**

Convert the following control flowchart to code:

city == “Pittsburgh”

isGoodMIT == True

print(“Carnegie Mellon University”)

print(“Central Michigan University”)

print(“Michigan Institute Technology”)

print(“Massachusetts Institute of Technology”)

True

True

False

college == “CMU”

True

False

False

Answer:

**CONDITIONAL CODE TRACE**

Answer:

def f(a, b, c):

 result = 'A'

 if (a>b):

 result += 'B'

 elif (b>c):

 result += 'C'

 if (a>c):

 result += 'D'

 else:

 result += 'E'

 result += 'F'

 return result

print(f(1,2,3), f(2, 3,1))

# **CIRCUITS AND GATES**

You are given the following circuit:



Fill out the truth table below (you may not need all rows):

|  |  |  |
| --- | --- | --- |
| x | y | output |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Write the equivalent Boolean expression demonstrated by the table and circuit:

Answer

Bonus:

Recreate the circuit online using <https://logic.ly/demo>