CS 1112: Introduction To Programming

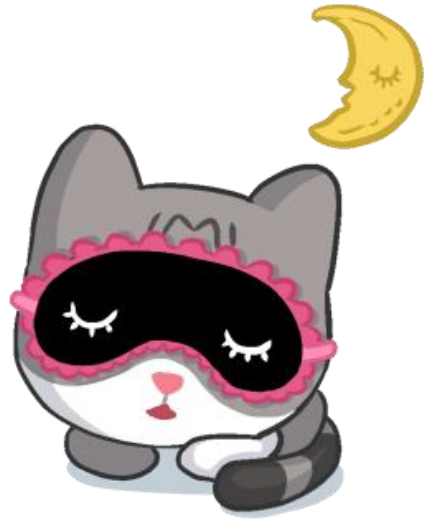
Pseudocode; input() function;

Summary of Vocabulary

Dr. Nada Basit // basit[*at*]Virginia[*dot*]edu

Friendly Reminders

- Your ***safety*** and ***comfort*** is important!
  - If you choose to wear a mask you are welcome to do so
  - *We will interpret wearing a mask as being considerate and caring of others in the classroom (not that you are sick), and realize that some may choose to mask to remain distanced*
- Remember to always be **kind, respectful, supportive, compassionate and mindful of others!** ☺
- Be an ***active*** participant in your learning!
  You're welcome and ***encouraged*** to ask questions during class!
- If you feel ***unwell***, or think you are, please stay home
  - *Contact us!* *We will work with you!*
  - Get some rest ☺
  - View the recorded lectures – *please allow 24-48 hours to post*



Announcements

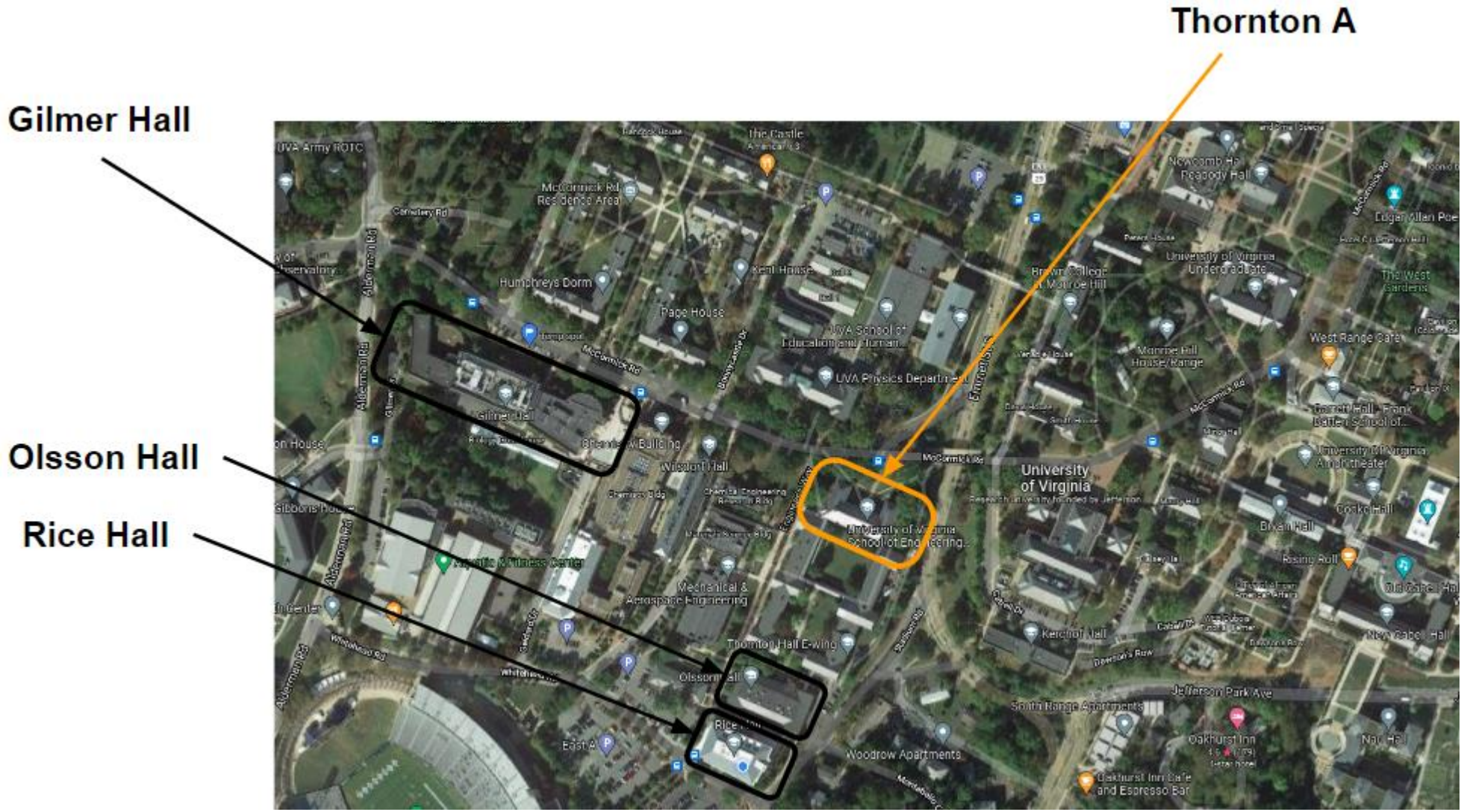
- **Programming Assignment: PA00 - Basics & Variable Types** was released Tuesday
  - PA-00 is due by 11:00pm on Wednesday, January 29
    - Submit on **Gradescope** (not on Canvas)
- **Quiz 1** will be released today (Friday)
  - Quiz 1 is due by 11:00pm on Monday, January 27
    - Find 30 minutes between Friday and Monday to take the quiz
    - No late quizzes accepted
    - No make-up quizzes allowed
    - Remember: we do drop your lowest two quiz scores
- Sherlock.cs.virginia.edu
  - Take quiz here! Details provided in the Quiz announcement
  - Please read these details carefully
  - Those with extra-time **SDAC** accommodations, this time will be built into the quiz

TA Office Hours

- TA office hours are in-person in **Thornton Stacks.** Location: Thornton A, 2nd floor
- Join the **electronic office hour queue** using the **link** that is posted on Canvas
  - TA office hours are **in-person**, but to join the queue, you use the **office hour queue tool**

- **Utilizing the Office Hours Queue:**
- Select our **course** (CS 1112)
- Click on blue "**Join Queue**"
- Enter relevant information:
  - **Subject**, **Description** (be descriptive, please!), and **Location** (in Thornton Stacks)
  - Wait time may be reduced if you choose to click on the **check-box** to be *grouped*
- Click on blue "**Join Queue**"
- Take the **quick survey** after each of your office hour interactions – ***we greatly appreciate your feedback!!***

TA Office Hours: Thornton Stacks



Reminder: Variables

- In Math:
  - A name that refers to a particular value
- In Statistics:
  - A name for a thing that has one value, but we don't know what it is
- In **Programming**:
  - A named **value** that can vary over the course of doing something
  - Python variable names: all **lower case**, must begin with a **letter** (a-z), *for example:*
    - `color`
  - Longer variable names: use underscore to join words, *for example:*
    - `table_length`
    - `num_transactions`

Pseudocode

"kind of" code

A steppingstone towards writing actual code

Pseudocode

- "*Pseudo*" means "*Not quite*"
- "*Code*" means *"write things in a way a computer can understand"*
- Pseudocode:
  - *Not quite* in the computer's language
- Good for expressing things unambiguously
  - But still using **English** (and simple symbols) to gloss over the *details*

Pseudocode: ignore most syntax, focus on algorithm

- *Not formal code*
  - Informal description of algorithm
- The **syntax** is informal, the **algorithm** is not!
  - It's still specific, detailed, and organized
  - Clearly indicates a **sequence** of actions the program will take
  - No specific syntax
- Can be translated into a programming language by adding the proper syntax

We Can Use Pseudocode For All of These:

- Sequence
  - A series of statements that execute *one after another*
- Condition (if)
  - Used to *decide* which of two more different statement to **execute** based on certain **conditions** (that is, “do this OR do that”)
- Repetition (loop)
  - To repeat statements *while* certain conditions are true
- Named Action (Subprogram or Function)
  - A small part of another program solving a certain problem

Reminder: Type Casting

Type casting: "reshaping" a value of one type into a value of a different type.

You can ask the compiler to tell you what data type something is by using the "type()" function. E.g., print(type(x))

- Type-casting is a method with which we can switch (*convert*) a variable's **type** to another
- i = 1
  f = 2.9
  s = "33"
  st = "4.5"
  b = "True"
- Conversions:

		Result:	
• float to int:	`int(f)`	`2`	
• int to float:	`float(i)`	`1.0`	
• str to int:	`int(s)`	`33`	[*invalid*: int("hello")] !!
• str to float:	`float(st)`	`4.5`	
• str to bool:	`bool(b)`	`True`	[*always True*: bool("hello") = True] !!

Obtaining user input: One of the most common situations where you would use type casting!

★ `input()` function: *receive user input (from keyboard)*

- You can use the input() function to ask a question to the user and receive input!
- This is very handy in many programs where the program requires some additional information before it can perform its tasks
- What you need:
  - A **variable** to hold the input (using an assignment statement) e.g., temp
  - The **input()** function – *takes as its parameter the question you want to ask-inside the ( )s*
  - A **prompt** (question) to the user `What is the temperature today?`
- For example:

```
temp = input("What is the temperature today? ")
```

equals symbol (**=**): *assignment statement*

  - The user types the answer into the console (on the screen) `What is the temperature today? 73`
  - The left-hand side of the "=" receives the value

★`input()` function: *receive user input (from keyboard)*

- **Notes**:
  - What ever input you type; it gets transferred and stored as a STRING

```
temp = input("What is the temperature today? ")
```

  - If you want the data to remain as a **string**, then there is nothing further you need to do
  - However, what if you ask for something like an "age" or a "temperature"? Most likely you want to do something with this number. You cannot do "math"-like things to the string "101.5" – it is *still* a string (THE OPERATIONS FOR NUMERICAL DATA DO NOT WORK ON **STRINGS**!)
    - You must **type cast** this value (to a float, for example) before you can use it as a number. There are two ways:

```
temp = float(temp)  # type cast to float from string
```

```
gpa = float(input("What is your GPA? "))
      # Type cast to a float immediately before storing in variable gpa
```

Quick & Fun Survey Questions

Get to know your peers! ☺

Digital Watch or Analog Watch?

Vocabulary (1)

- **Statement**: A single action to perform (one line of pseudocode)
  - `Set X to be x / 2`
- **Execute**: Follow an instruction or sequence of instructions ("to run")
  - `Set X to be x / 2`
  - When the computer *actually* does this line
- **Program**: the sequence of statements to be executed
- **Conditional**: Code that is only *sometimes* executed
  - A condition (your *compid* number is even)
  - Statements that you execute when the condition is **true**
  - Statements that you execute when the condition is **false** (*optional*)

```
if (condition):
    statements to execute if condition is true
otherwise:
    statements to execute if condition is false
```

- **Keyword**: a word that guides how to execute your program, e.g., **if**

★Vocabulary (2)

- **Variable**: a name for a value, the *value* it is naming may *change* during execution - the name cannot be a **keyword**
- **Expression**: a portion of a statement describing a value
  - `Set X to be x / 2`
- **Loops**: statements that we might execute over and over again

```
Repeat as long as x is not equal to 2:
    statements to be repeated a # of times based on the condition
    statements to be repeated a # of times based on the condition
```

- **Functions**: functions, methods, subroutines – *named actions*
  - We have ***used*** and ***seen*** functions already! *We haven't written/defined functions ourselves, yet*

```
Type casting:  int(x)          this is using a function "int" that takes in one parameter (the item to be type casted)
User input:    input("What is your name? ")   the prompt/question is posted on the screen and the answer is captured
Printing:      print("The result is: " + str(x))   we have seen/used the print function a lot already!
```

Example - pseudocode

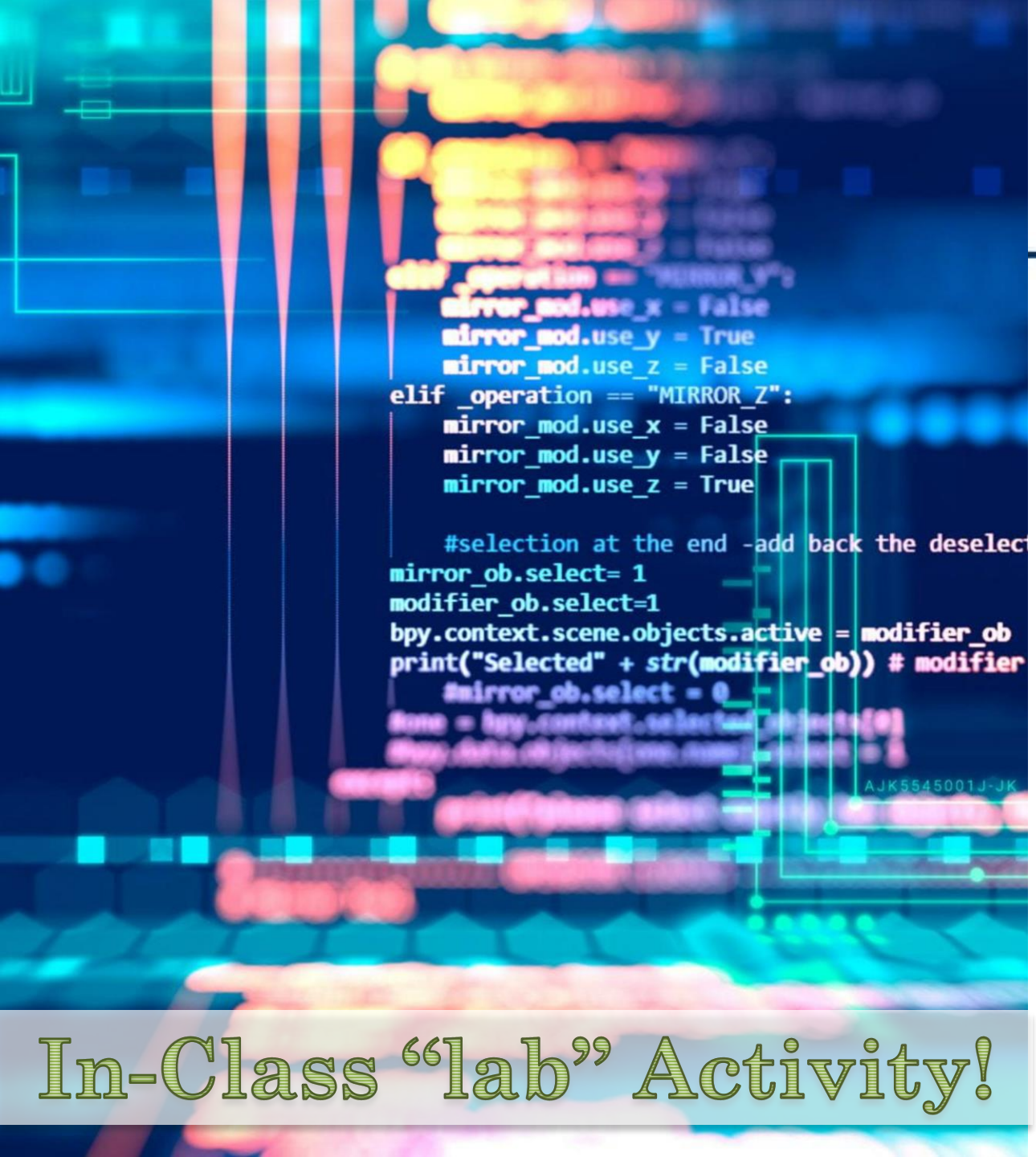
- Prompt the user for the day of the week
- Store the result in a variable named "day"
- Prompt the user for the current temperature (in *F)
- Store the result in a variable named "temp"
- Type-cast the value into a float
- If the day is ("Saturday" or "Sunday") AND if the temperature is > 75 degrees
  - Display (print) "A lovely day for a walk"
- Else
  - Display (print) "Stay inside and read a book"

Python Demonstration

Let's jump on PyCharm!

pseudocode_user_input.py

pseudocode_in_class.py (if time)



In-Class "lab" Activity!

Use:
pseudocode_activity_ica.py File

- In pairs or groups up to three work on the following questions
- **Convert code into pseudocode**
- **Convert pseudocode into code**
- *Do your best to be as precise as possible!*

Remember to check-in with a TA before leaving class today!

Remember to check-in with a TA and show them your solutions to the three (3) activities before you leave class today!

Pseudocode In-Class "lab" Activity

-- ACTIVITY #1 --

1. Prompt the user to enter the first number.

2. Store the entered number in a variable called num1.

3. Prompt the user to enter the second number.

4. Store the entered number in a variable called num2.

5. **Add** num1 and num2 and store the result in a variable called sum.

6. Display the sum.

Think: print

Remember to check-in with a TA and show them your solutions to the three (3) activities before you leave class today!

Pseudocode In-Class "lab" Activity

-- ACTIVITY #2 --

1. Prompt the user to enter a number.

2. Store the entered number in a variable called num.

3. If num is greater than 0, display "The number is positive."

4. If num is less than 0, display "The number is negative."

5. If num is equal to 0, display "The number is zero."

Think: print

Remember to check-in with a TA and show them your solutions to the three (3) activities before you leave class today!

Pseudocode In-Class "lab" Activity

```
# -- ACTIVITY #3 -- CODE TO PSEUDOCODE

num1 = float(input("Enter the first number: "))

num2 = float(input("Enter the second number: "))

num3 = float(input("Enter the third number: "))


total = num1 + num2 + num3

average = total / 3


print("The average is:", average)
```

Notes/Reminders…

Reminder: CS Laptop Loaner Program

- This course requires students to have a **laptop**
- I realize that not everybody might have one (nor necessarily need one for their desired major / path…)
- If you do not have a laptop for any reason… ***not to worry!***
- The CS department's Systems staff has a notebook / laptop loaner program and will be able to loan you a notebook / laptop computer for the duration of the semester if you don't have one or if you cannot afford one.
  - Also available if your laptop is broken and under repair, we can arrange for you to receive a loaner laptop for a week or two until your own laptop is fixed

Interested? Link: https://www.cs.virginia.edu/wiki/doku.php?id=cs_laptop_loaner

I am happy to be your sponsor. Please let me know.

Tools: Piazza

- We will use **Piazza** in the following way:
  - Website: https://piazza.com/ [Linked through Canvas]
  - Piazza is a great tool for asking questions about **course content**, **policies**, or getting help on **homework** assignments
  - While you are waiting for an answer, see if there's an answer you can provide to someone else's question. We're all in this together! CS is a team sport! ☺
  - TAs will monitor and answer questions throughout the semester
  - Not a means to help you debug your code! (See more below)

It is very important to remember the following:

- **Do not post** complete or partial code solutions (for Homework) on Piazza when seeking answers to your question unless it is in a **PRIVATE** post
- **Do not post** complete or partial quiz solutions (code or short-answer) when seeking answers to your question unless it is in a **PRIVATE** post

Tools: Gradescope

- We will use **Gradescope** in the following way:
  - Website: https://www.gradescope.com/
  - Linked through Canvas
  - Homework assignments will be **submitted**
    - Most programming assignments are autograded
    - Some aspects of programming assignments may be manually graded