COGS 18 Exam #1 (Answer Key)

Fill out your Name and PID here:

Name: _____

PID: _____

Do not begin until instructed by Professor Ellis to do so.

Exam Notes:

- Put your PID at the top of each page.
- This is a closed book test. You may not use any resources other than your own brain and your writing utensil.
- All work should be your own. Keep your eyes on your own exam.
- If you are unsure of what any instructions mean, raise your hand to ask a TA, IA or the Professor.
- You'll have until 12:20 to complete the exam.
- Answer all questions - partial credit is possible.
- Your exam should have 6 pages.
- There are 9 possible points for the in-person exam (3.5 points for the take-home portion).

PID: _____

Multiple Choice: Circle the correct response. **(1pt each)**

1. During variable assignment, we _____ use the _____ operator:
   - **A. always; =**
   - B. sometimes; =
   - C. always ; ==
   - D. sometimes; ==

2. In a conditional `else` is used when…
   - A. you want to specify the condition under which the else code block should execute
   - B. **you want to specify what code should execute if the `if` and `elif` conditions all evaluate as `False`**
   - C. you want to be able to specify input parameters, carry out operations on that input, and return some output
   - D. you want to specify a code block that always executes no matter what.

3. If you wanted to define a function that would take a collection as input and update a subset of the elements of that collection within that function, you would want the input collection to be a(n)…
   - A. Tuple
   - B. Integer
   - C. Boolean
   - **D. List**

4. When you want to create a variable that allows you to store key value pairs, which of the following braces would you utilize:
   - A. []
   - **B. {}**
   - C. ()
   - D. ||

5. If you wanted to use a conditional that would have the code block after the `if` statement execute, which of the following conditionals would you have following `if`?
   - A. `'Cogs18' == 'COGS18'`
   - B. `20 > 100`
   - C. `2 * 10 == 12`
   - **D. `10 <= 10`**

6. You've defined a function '`do_math`' that takes a single parameter as input. Inside the function your code takes the input parameter, multiplies it by 2 and then subtracts 2 from that value, ultimately returning the result of these operations from the function. How would you execute your function so that it would return the value 18?
   - A. do_math(10*2-2)
   - B. do_math[10*2-2]
   - **C. do_math(10)**
   - D. do_math[10]

PID: _____

Short Answers:

7. *Math Operators* - Write out how each expression will evaluate: **(0.25pt each)**

`(4 - 2) / 2`	_____1_____
`17 % 5`	_____2_____
`3 ** 2`	_____9_____
`3 * 2`	_____6_____
`'love'  +  'cogs18'`	_____lovecogs18_____
`2 + 'python'`	_____Error_____

8. *Comparison and Boolean Operators* - Write out how each expression will evaluate:**(0.25pt each)**

`True or False`	_____True_____
`7>10 and 3<4`	_____False_____
`7>10 or 3<4`	_____True_____
`3 == 8 or 20 != 20`	_____False_____

9. *Membership Operators* - Given the following two variables, write out how each expression will evaluate: **(0.25pt each)**

```
TAs = 'Annapurna, Pooja, William'
IAs = ['Ethan', 'Jiayi', 'Josh', 'Lillian', 'Mingson']
```

`'Will' in TAs`	_____True_____
`'Pooja' in TAs`	_____True_____
`'Ethan' in IAs`	_____True_____
`'Ming' in IAs`	_____False_____

10. Variables and Math operators - Given the following code, what would be the output? **(0.5pt)**

```
val_a = 2
val_b = 6
val_c = (3 + 2) * (val_b/val_a)
print(val_c + 2)
```

Your answer: _____17_____

PID: _____

CONTINUE TO THE NEXT PAGE

11. Given the following variable `cogs18`, describe 1) the type of variable generated and 2) how/what information is stored within the variable. **(2pts)**

```
cogs18 = {'Assignment' : 8,  'Midterm' : 12.5, 'Lab' : 2}
```

1) Dictionary
2) Strings containing the grading category are the **keys**; **ints/floats** are the corresponding **values**

12. What is the difference between function definition and function execution? **(2pts)**

Defining: instructions that will be carried out
Executing: carrying out the specified instructions on a given input (actually use the instructions)

13. What is an example of a time when it would be helpful to write a function in python? Describe why a function would be particularly useful in that scenario. **(4pts)**

Demonstrates function understanding (1pt)
Provides a good example (1pt)
Ties example to aspects of function (2pts)

Note: if you gave an example, but failed to mention aspects of functions/inputs/operations/returns, 2-3 points were deducted

CONTINUE TO THE NEXT PAGE

PID: _____

Concepts (that you will implement technically on the take-home portion)**:** *For these next three questions, each question will be related to the same task: writing code to make a very basic memory game. The idea here is that there will be three functions: 1) a function that displays a single character; 2) a function that displays a set of characters; 3) a function that determines if the single character is in the set of characters.*

14. **Reading Code & Function Execution**: You sit down to write your first function (`display_char`) and get it right on the first go! a) Describe what this function accomplishes, b) write the code you would use to execute this function to `return` 'Ʀ' from the function, and c) explain a scenario where this function, as written, would error.

```
def display_char(index):

    objects = ['x', 'ã', 'ţ', '%', '§', '¶',
               'Ł', 'Œ', 'Ɓ', 'Ʀ', 'Ɏ', 'ç']

    char = objects[index]

    return char
```

1) Function description:
- **Takes index as parameter/input (0.5)**
- **Defines list (0.5)**
- **Indexes into list (0.5)**
- **Stores character in char (0.25)**
- **Returns character (0.25)**
-

2) Function execution: _____**`display_char(9) or display_char(index=9) OR display_char(-3) or display_char(index=-3)`** _____ **(1pt)**

3) When it would error: **Would error when an index outside the list were provided: 12+ (or if something other than an integer were provided as input) (1pt)**

CONTINUE TO THE NEXT PAGE

PID: _____

15. **Debugging**: You sit down to write a function (`display_objects`) that will return a slice of a specified list (`objects`). The idea is that, when executed, the function would use the values specified for the three parameters (`start`, `stop`, and `step`). However, when you sit down to write your function, you draft the following. It's not quite doing what you want yet… Describe 1) what the output from this function as written would be and 2) what has to be changed for this code to accomplish what we were hoping it would accomplish.

```
def display_objects(start, stop, skip):

    objects = ['x', 'ã', 'ţ', '%', '§', '¶',
               'Ł', 'Œ', 'Ɓ', 'Ŗ', 'Ɏ', 'ç']

    objects[start:stop:skip]

    return objects
```

a) Output as written **(1pt)**
As written, this function would return the full list (objects) defined within the function

b) Edits needed to accomplish goal **(1pt)**
The slice has to either be assigned to a variable and that variable has to be returned from the function or the slice would have to be returned directly

16. **Function Planning**: Finally, after you get `display_objects` and `display_char` to work, you're ready to put everything together with the function `determine_match`. This function should take in two parameters, the output from `display_char` and the output from `display_objects`. If the output from `display_char` is in the output from `display_objects`, this function should return True; otherwise, it should return False. Write some pseudocode to describe the components of the function you would need to accomplish this task. **(5pts)**

```
def determine_match(char, list_of_objects):

    if char in list_of_objects:
        output = True
    else:
        output = False

    return output
```

Pieces for full credit: 1) function name/mention of a function; 2) input parameters (2 of them); 3) conditional structure; 4) conditions mentioned; 5) values that would be returned under each condition