**Test 1 CS 115 Fall 2018**

Name (Print please) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Score 210 - \_\_\_\_\_\_\_\_\_\_\_ = \_\_\_\_\_\_\_\_\_\_\_\_\_ + \_\_\_\_\_\_\_\_ Bonus = \_\_\_\_\_\_\_\_\_\_\_\_\_ / 210 points

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **01** M 9 am Kelsey | **02** M 10 am Maniell | **03** M 11 am Steven | **04** M 12 pm Maniell | **06** M 3 pm Yujiro | **401 5** pm Matthew |

**Please circle your section!** This is a closed note, closed book exam. No electronic devices of any kind may be used during the exam, **including calculators.** **Please turn off any phones, etc.** No looking at other papers is allowed. No communication with other people during the exam is allowed. No baseball caps may be worn during the test. You should have signed a document that you promised to do your own work on tests (Academic Honesty Form). If you did not, you will not get credit for this exam. If you have a question, REMAIN IN YOUR SEAT, raise your hand and wait for a proctor to come to you. Write your answers on this paper. If you need more space, use the back of a page - and put a note to that effect on the front! Scratch paper is not allowed.

When you are finished, bring your test and your Picture ID to the front of the room. Sign the roster there as proof that you were present.

**If a result is a floating-point number, make sure you show a decimal point. String constants should look like strings. Write out Boolean constants as words, not as T or F. ␢ should be used as a space.**

1. (4 points) Turing had the idea for a Universal Machine which could be programmed by putting **what** into RAM?

a. data b. program instructions c. numbers d. images

2. (4 points) What does it mean to be case sensitive?

a. uppercase letters are not allowed in the language

b. all keywords are in lowercase

c. lowercase letters and uppercase letters are treated as different symbols

d. the identifiers DOG and dog are treated as the same identifier.

**(2 points) Is Python case sensitive?**

3. (6 points) Which is faster data access? RAM or a hard drive (circle one)

 Which is cheaper per Megabyte? RAM or a hard drive (circle one)

 Which is volatile? RAM or a hard drive (circle one)

4. (5 points) You have a memory stick with a capacity of 32 Gigabytes. You have some picture

files which are each 256 megabytes. How many of those picture files can you store on the memory stick at one time?

5. (4 points) Charles Babbage was associated with

 a. Ada Lovelace b. Grace Hopper c. Alan Turing d. Isaac Newton

6. (5 points) The translator can help you find syntax / semantic errors. (circle one) **How do you**

**find the other type of error?**

7. (4 points) A translator program which translates one line of code then executes it, and repeats

that cycle until the program is done is known as an interpreter / a compiler. (circle one)

 Which one is Python as we use it in this class?

8. (8 points) Circle the INvalid identifiers. For each one that is INvalid, state the reason why.

 a. 2smart4u b. Angle1 c. def d. long\_one e. a?b

9. (5 points) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ testing says that when you change the

code, you must redo all test cases.

10. (12 points) Fill the table with the value and type of each expression

|  |  |  |
| --- | --- | --- |
| **Expression** | **Value** | **Type** |
| A. 23 // 4 |  |  |
| B. 170 % 10 |  |  |
| C. 3 \*\* 2.0 |  |  |
| D. 17 / 5 |  |  |

11. (6 points) In this expression below, which **operator** gets done first? \_\_\_\_\_\_\_\_\_\_\_\_\_\_

 second? \_\_\_\_\_\_\_\_\_\_\_\_\_\_ last? \_\_\_\_\_\_\_\_\_\_\_\_\_

 **b > 9 or 7 < c + 4 and d == f**

12. (5 points) If **first** is 12 and **second** is 3, what is **first**'s value

 after this statement, **first \*= second + 1,** is executed? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

13. (9 points) What is the output of the code below? You may or may not need all the lines. ␢ should be used as a space where needed.

 print("abc", "def", sep="\*")

 var = 23

 print(var, "var", end="?")

 print("End")

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

14. (8 points) Write an import statement that can be used with the code below:

 **print(sqrt(49))**

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

15. (9 points) Fill in the table for the code below so that all branches are executed at least once.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| if var1 == var2: if var1 > 15: print("yes") else: print("maybe") |

|  |  |
| --- | --- |
| **var1, var2** | **Output** |
|  |  |
|  |  |
|  |  |

 |

16. (4 points) What is the value of this expression? **2e4 + 1e3** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 What is its type? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

17. (4 points) If an expression is a tautology, what is its value? \_\_\_\_\_\_\_\_ its type? \_\_\_\_\_\_\_\_\_\_\_

18. (6 points) What is the output of this code? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 ␢ should be used as a space as needed.

 var1 = 20

 var2 = var1 + 4 # var2 = 29

 var3 = var1

 var1 = 15

 print (var1, var2, var3)

19. (8 points) Write this equation as a Python assignment statement. Assume this is regular

division, not floor division.

 $var1= \frac{4varA+varB}{varC} $

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

20. (5 points) What is the value of this expression? **temp == 2 or 5** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 **Why is it that value?**

21. (8 points) Name the columns of a complete test plan, as used by a software company.

22. (4 points) What does it mean to put \ at the end of a line of Python code?

23. (6 points) Name two escaped characters and describe what they mean.

24. (8 points) Fill in this truth table for the **or** operator. Write out the Boolean constants, not just initials.

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **A or B** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

25. (6 points) Name 4 properties of a variable. Circle the properties you named which can change

 during the course of a program in Python.

26. (6 points) What data type does input() return? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 What data type does print() return? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

27. (8 points) How many times do these loops iterate?

 for k in range(5, 13, 3): \_\_\_\_\_\_\_\_\_\_\_ for m in range(23): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 for n in range(10, 14): \_\_\_\_\_\_\_\_\_\_\_\_ for j in range(15, 3): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

28. (12 points) Fill in the table with values and types of the expressions.

**If a result is a floating-point number, make sure you show a decimal point. String constants should look like strings. Write out Boolean constants as words, not as T or F.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Expression** | **Value** | **Type** |
| a. | "joN" < "joe" |  |  |
| b. | x > 10 and x < 0 |  |  |
| c. | int(7.99) |  |  |
| g. | round(10.713, 2) |  |  |

29. (6 points) Write a statement that will output to shell a random integer between 1 and 12 inclusive. Assume the proper library is imported.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

30. (25 points) Write a complete program that has a main function which solves this problem:

The user inputs a number of ounces. The program outputs the number of pounds and ounces in that amount. There are 16 ounces in a pound. The output should be grammatically correct, that is, taking into account singular and plural forms (“ounces” vs. “ounce”, “pounds” vs. “pound”). You can assume the number of ounces from the user is an integer. Match the text of the sample runs exactly.

You have to put in one comment at the top with your name in it. You don't have to document any more than that. Write your logic as efficiently as you can; do not have redundant tests.

**Sample run:**

How many ounces? 33

You have 2 pounds and 1 ounce.

**Sample run:**

How many ounces? 15

You have 0 pounds and 15 ounces.

**Sample run:**

How many ounces? 18

You have 1 pound and 2 ounces.

**Sample run:**

How many ounces? 17

You have 1 pound and 1 ounce.

**Bonus (4 points)** What does Guido van Rossum have to do with computer science?