CS303E Mock Exam 1
Dr. Bill Young § Spring, 2025

Name: _____ EID: _____

Read the questions carefully, and answer each question in the space provided. If you like, you can use scratch paper to do your work, but copy your answers neatly and legibly onto the test paper. Only answers recorded on the test paper will be graded. Don't write in the spaces marked "Page Total" at the bottom of each page.

1. (10 points: 1 point each) The following are true/false questions. **Write either T or F in the boxes at the bottom of page 1.** If there's any counterexample, it's false.

   (a) To run a Python file in batch mode, you will need to import the corresponding module from within the Python interpreter command loop.

   (b) `_3rd_` is a valid variable name.

   (c) If `x` is a floating point number, then `int(x)` will return `x` rounded to the nearest integer.

   (d) The values `3` and `"3"` are equivalent representations of the number three, i.e. after the assignments `x = 3` and `y = "3"`, `x` and `y` would hold exactly the same value.

   (e) Basic arithmetic operations on floats like addition, subtraction, multiplication, and division are only approximate in python, meaning that for certain calculations the results may have some small error.

   (f) There is no way to include a backslash (i.e. this character → \) in a string, since the backslash is reserved for escape sequences.

   (g) If `x` is a variable holding a float, then the statement `print(round(x, 2))` will display the value of `x` with exactly two digits after the decimal point.

   (h) The expression `bool(None)` is equivalent to `False`.

   (i) If you have an if-elif statement, then you must also have a corresponding else branch.

   (j) If `a` and `b` are booleans, the boolean expressions `not (a and b)` and `not a and b` may not evaluate to the same value.

a	b	c	d	e	f	g	h	i	j

Page total: _____/10

2. The following questions (worth 1 point each) require you to evaluate a Python expression in the left hand column. For each question, write what the expression evaluates to on the provided line. If evaluation results in an error, write "error"; you don't have to identity the specific type of error. You may assume the math library has been imported.

   You must show a value of the appropriate type. For example, `7.0` **rather than** `7` **for a float and** `"7"` **instead of** `7` **for a string. Answers that do not indicate the data type correctly are wrong.**

   (a) `(ord('z') - ord('y')) * -4 + 2` _____

   (b) `0 // 8 + 4` _____

   (c) `math.ceil(2.718)` _____

   (d) `"Hello" + "World"` _____

   (e) `round(3.14159, 3)` _____

   (f) `(3 % 100) and not (100 % 10)` _____

   (g) `"56" + "4"` _____

   (h) `int("2.5")` _____

   (i) `60 / 3 + 4 * 2` _____

   (j) `float(10)` _____

Page total: _____/10

Questions 3–7 are multiple choice. Each counts 2 points. **Write the letter of the BEST answer in the boxes provided. Please write your answer in UPPERCASE. Each problem has a single answer.**

3. Which type of error, if any, is present in the following code?

```
import math
x = -1
print("The number", format(x, "0.1f") + " has", \
    "only complex square roots")
```

   A. syntax error
   B. runtime error
   C. logic error
   D. no error

4. Suppose `a` and `b` are integers. Which of the following expressions won't result in the comparison `(a < b)` being evaluated?

   A. `True and (a < b)`
   B. `True or (a < b)`
   C. `not (a < b)`
   D. all of these result in the comparison being evaluated

5. Which of the following is NOT an immutable type in python?

   A. `int` B. `float` C. `str` D. all of these are immutable

6. Assume that `a` is an integer. Three of the following boolean expressions are equivalent regardless of the value of `a`. Which one is not equivalent to the others?

   A. `(a < 10) or (a > 50)`
   B. `not (10 < a < 50)`
   C. `(50 < a) or (10 > a)`
   D. `not ((a <= 50) and (10 <= a))`

7. Which of the following operators has the highest precedence?

   A. `+` (addition) B. `*` (multiplication) C. `**` (exponentiation) D. `%` (mod)

3	4	5	6	7

Page total: _____/10

Questions 8–10 (2 points each) require you to trace the behavior of some Python code and identify the output of that code. For each question, write the output for the code segment in the provided box. If executing the code gives an error, write "ERROR" in the box; you're not required to identify what type of error occurs. *Don't worry about whether it goes to the next line at the end.*

8.
```
print("Greetings", "I have" + "come from", "a different", \
    "planet", sep="$", end="!!!")
```

9.
```
# a similar type of bowling handicap
max = 300
multiplier = 0.5
average = 240
handicap = int((max - average) * multiplier)
handicap = max(handicap, 0)
print(handicap)
```

10.
```
x = -2
y = 3
a -= x
b *= y
print(x, y)
```

Page total: _____/6

11. (1 point each) The following questions require you to write a *Python expression* that returns the indicated value. You can assume that any modules you need have been imported. Note that this asks for a single, one-line expression for each question, not a longer program fragment. So, you should not have any assignments, loops, or if statements.

    (a) _____
    the result of a standard die roll (i.e. a uniformly random integer between 1 and 6, inclusive)

    (b) _____
    for a given integer `n`, a boolean indicating whether `n` is divisible by 3 but not 5

    (c) _____
    for a given integer `n` having at least two digits, the digit in the tens place in `n`, i.e. the second digit from the right

    (d) _____
    for given floats `a`, `b`, and `c`, the average of these three values

    (e) _____
    for given floats `a`, `b`, and `c`, the range of these three values, i.e. the difference between the largest and smallest value

    (f) _____
    for given integers `x` and `y`, the remainder when `x` is divided by `y`

    (g) _____
    a user-provided input, converted to a floating-point value, after prompting the user with `"Enter a number"`

Page total: _____/7

12. (10 points) Write code you might put into file `getQuadrant.py` to accept from the user two float values representing the x and y coordinates of a point in the Cartesian plane. You don't need to validate the inputs. The code should print which quadrant the specified point is in. Recall that the plane is divided into four quadrants: quadrant 1 is the upper right quadrant, quadrant 2 is the upper left quadrant, quadrant 3 is the lower left quadrant, and quadrant 4 is the lower right quadrant. If the given point lies on the x-axis and/or y-axis, the function should return 0. Here is some sample behavior:

```
> python getQuadrant.py
Enter x: -2.5                          # User input
Enter y: -3.0                          # User input
Quadrant: 3                            # Printed by program
> python getQuadrant.py
Enter x: -0.5
Enter y: 0.0
Quadrant: 0
> python getQuadrant.py
Enter x: 3.0
Enter y: -5
Quadrant: 4
```

```
# Your code here:
```

Page total: _____/10

13. (10 points) You're a delivery driver, and you have the following agreement with your employer: whenever a customer tips less than a dollar, the company pays you the difference between the given tip and a dollar. For example, if the customer only tips you 25 cents, then the company will pay you an additional 75 cents for that delivery. In exchange, whenever a customer tips more than five dollars, you only get to keep five dollars of it, and the rest must be given to the company. For example, if a customer tips you $8.75, you must pay $3.75 of that to the company.

    Write code you could put into file `deliveryTips.py` that will accept from the user a dollar tip amount and **print the tip amount the company pays you**, not what you keep from the customer. If you'd have to give money back to the company, put the amount in parentheses. Display the value with two digits after the decimal point, but you don't need to show a dollar sign.

    ```
    > python deliveryTips.py
    Customer tip: 8.75
    (3.75)
    > python deliveryTips.py
    Customer tip: 0.25
    0.75
    > python deliveryTips.py
    Customer tip: 4.00
    0.00
    ```

```
# Your code here
```

Page total: _____/10