Name: ______________________________

Email Address (UW Net ID): ______________________________@uw.edu

CSE 160 Winter 2023 - Midterm Exam

Instructions:

- You have **45 minutes** to complete this exam.
- The exam is **closed book**, including no calculators, computers, phones, watches or other electronics.
- You are allowed a single sheet of notes for yourself.
- We also provide a syntax reference sheet.
- Turn in *all sheets* of this exam, together and in the same order when you are finished.
- When time has been called, you must put down your pencil and stop writing.
    - **Points will be deducted if you are still writing after time has been called.**
- You may only use parts and features of Python that have been covered in class up to this point.
- You may ask questions by raising your hand, and a TA will come over to you.

Good luck!

Question	**Points**
Question 1	6
Question 2	6
Question 3	6
Question 4	4
Question 5	5
Question 6	4
Question 7	5
TOTAL	**36**

1) [6 pts] For each of the below expressions, write what the expression evaluates to and the type of that value. You should assume that variables have been declared and assigned as follows:

```
a = 11
b = "hello"
c = False
d = 10.5
e = 2
f = { "a": 1, 1: "a", "b": b }
```

Expression	Value	Type
`a / e`		
`e + a`		
`a % 2 == 0 or not c`		
`not (len(b) < 5 and d > e)`		
`b[1] + str(d * e)`		
`f["b"][2] + f[1]`		

2) [6 pts] Given a file named **`cool_story.txt`** (you should assume this path points to a valid file location and that the file has content), the code below aims to count the number of the common words "a", "the", and "and". Write the body of the for loop below such that **`common_words[0]`** contains the number of occurrences of "a", **`common_words[1]`** contains the number of occurrences of "the", and **`common_words[2]`** contains the number of occurrences of "and".

```
file = open("cool_story.txt", "r")
common_words = [0, 0, 0]

for line in file:




file.close()
```

3) [6 pts] Each of the code snippets below has a different error. For each snippet, write one sentence naming and describing the error. Then, rewrite the code to correctly produce the indented output (indicated in the comment above the snippet).

A.
```
# The expected output is: 60
height = [55, 65, 77, 56, 63, 74, 57, 60]
print(height[8])
```

B.
```
# The expected output is:
# I love Python!
# I love Python!
# I love Python!
for i in range(3):
print("I love Python!")
```

C.
```
# The expected output is: I don't want to listen to music now.
print('I don't want to listen to music now.')
```

4) [4 pts] For each of the following for loops, write in the box to the right how many times "`I love Python!`" is printed. Your answer should be a number.

```
for i in range(5):
    print("I love Python!")
```

```
for i in range(4, 9):
    print("I love Python!")
```

```
for i in range(11, 11):
    print("I love Python!")
```

```
for i in range(4, -9, -3):
    print("I love Python!")
```

```
for i in [1, 3, 5, 19393, -3]:
    print("I love Python!")
```

```
for i in [[1], [2, -1], [-3, 11, 64], [1, [32]], 5]:
    print("I love Python!")
```

```
for i in range(1, 19, 3):
    if i % 5 == 2 or i % 4 == 0:
        print("I love Python!")
```

```
numbers = [3, 2, 1, 5, 7, -4]
for i in numbers[2:5]:
    print("I love Python")
```

5) [5 pts] A Biology student has started learning Python to help with their research. They have written the function below.

```
def count_num_of_insects(animal_data):
    '''count_num_of_insects is designed to count the amount of insects in
       animal_data solely by counting how many have six (6) legs.

    Inputs: a list of integers where each number is an animal and how many
            legs it has. For example, for three animals where the first two have
            four legs and the third has 6, the input would be [4, 4, 6].
    Outputs: an integer that shows the amount of insects in the sample.
    '''
    insect_num = 0
    for animal in animal_data:
        if animal == 6:
            insect_num += 1
```

Later they call this function in their code with:

```
print(count_num_of_insects([4, 4, 6]))
```

The student has found that their code is not giving them the desired output.

A. Write one to two sentences on why the code is not producing the desired output:

B. What is the output of the above code?

C. What would they need to change in their function to give their desired output?

6) [4 pts] Grace wants to set her alarm clock for the week. To do this, she wants you to write a function named **`set_alarm`** that returns the time (int) she should set her alarm clock for.

The function takes in the **`day`** (string) of the week and whether she is on vacation or not, **`is_vacation`** (bool, as in True or False). You may safely assume that **`day`** is *always* a valid day of the week ("**`Monday`**", "**`Tuesday`**", "**`Wednesday`**", "**`Thursday`**", "**`Friday`**", "**`Saturday`**", "**`Sunday`**") and that **`is_vacation`** is *always* a boolean.

If it is a weekday ("**`Monday`**", "**`Tuesday`**", "**`Wednesday`**", "**`Thursday`**", or "**`Friday`**"), she wants to set her alarm clock for **`8`**. On weekends she wants to set her alarm clock to **`10`**. But, if she is on vacation, she always wants her alarm clock to be set to **`11`**, regardless of the day of the week. Implement the function that returns the correct time.

Example calls and return values:

- **`set_alarm("Monday", False)`** should return **`8`**
- **`set_alarm("Sunday", False)`** should return **`10`**
- **`set_alarm("Wednesday", True)`** should return **`11`**

```
def set_alarm(day, is_vacation):
    # Write your code below this line.
```

7) [5 pts] The highway patrol police has collected some data on traffic speed and needs help on analyzing it. Write a function **`average_speed_over_limits(speed_data, limits)`** that takes in a list of vehicle speeds (in miles/hr) and a list of speed limits (in miles/hr), then returns the average speed for vehicles that have exceeded those speed limits.

For example, **`average_speed_over_limits([60, 80, 53, 68], [60, 20])`** should return the list **`[74, 65.25]`** because

(1) the average speed of the two vehicles (at 80 miles/hr and 68 miles/hr) that exceeded 60 miles/hr is 74 miles/hr,
(2) all four vehicles have a speed exceeding 20 miles/hr and the average is 65.25 miles/hr.

If there is no vehicle data in **`speed_data`** or if there are no vehicles that exceed the speed limit in the data, it should set the average speed to 0 for the corresponding limit. Your function should return a list that has an average speed corresponding to each speed limit. You can assume that the speed limits and the vehicle speeds will all be positive integers.

```
def average_speed_over_limits(speed_data, limits):
    # Write your code below this line.
```