

Name: _____

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

Section: _____

CSE 160 Spring 2023 - Midterm Exam

Instructions:

- You have **40 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.
- All questions assume Python version 3.7, as we have been using all quarter.
- 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	4
Question 4	3.5
Question 5	6
Question 6	2
TOTAL	27.5

1. **[6 Pts]** For each of the expressions below, write what value it evaluates to and the type of that value (float, int, string, or bool). You should specify any floating point values only to one decimal point (e.g., 0.5)

Expression	Evaluation	Type
<code>2 ** 3 + 2</code>	10	int
<code>1 / 2</code>	0.5	float
<code>3 < 3 or False</code>	false	boolean
<code>1 // 2</code>	0	int
<code>3 % 2 + 1.0</code>	2.0	float
<code>"The CSE 160 TAs are " + 100</code>	Error	

2. **[6 pts]** There are two parts to question two, 2.1 and 2.2. Though similar, they are separate problems and the code from one does not affect the other.

2.1: Given the following program, what will be printed out when the program is run?

```
num = 5

def one(num):
    val = six(num) * 100
    return val

def six(val):
    if val % 5 > 3:
        val = zero(num) % 6
        return val
    elif val % 2 == 0:
        return zero(val * 2)
    else:
        return zero(num)

def zero(num):
    val = 3
    if num > 5:
        return 7
    else:
        return num * val

print(one(1))
print(one(6))
print(one(0))
```

```
1500
700
0
```

2.2: Given the following program, what will be printed out when the program is run?

```
def letter_code(num1, num2):  
    if num1 % 2 == 0:  
        return "A"  
    if num1 + num2 > 10:  
        return "B"  
    elif num2 % 2 == 1:  
        return "C"  
    else:  
        return "D"  
  
result1 = letter_code(6, 5)  
result2 = letter_code(1, 2)  
result3 = letter_code(5, 7)  
  
print(result1)  
print(result2)  
print(result3)
```

A
D
B

3. **[4 pts]** Below each code snippet, write what the output of the code is. Partial credit will be given for showing the calculation process to the right of the code snippet.

Part A:

```
sum = 0
for i in [1, 2, 3]:
    for j in [4, 5, 6]:
        sum = sum + j
print(sum)
```

45 (show work: $4+4+4+5+5+5+6+6+6=45$)

Part B:

```
a = [100, 200, 300]
b = a
c = a[0:2]
a.append(600)
print(b)
print(c)
```

[100, 200, 300, 600] (show work: $b = [100, 200, 300]$, changing on variable a is also changing on variable b)

[100, 200] (show work: writing out syntax on the variable a)

Part C:

```
j = 0
for i in [1, 2, 3]:
    for j in [4, 5, 6]:
        j = j + j
print(j)
```

12 (show work: $6+6 = 12$)

4. **[4 pts]** Write a function, `square_function(arr)`, that takes in a `list`, then modifies and returns it. At each index `i`, if `i` is even then the function must replace the value with the square of `i`. If `i` is odd, the value at that index should not be changed. For example:

```
arr = [0, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
square_function(arr) # returns [0, 0, 4, 0, 16, 0, 36, 0, 64]
```

```
def square_function(arr):  
    # Write your code below this line  
    for i in range(len(arr)):  
        if i % 2 == 0:  
            arr[i] = i * i  
    return arr
```

5. **[6 pts]** A concert that the CSE 160 TAs wanted to attend was sold out in under three minutes. They have gotten all the data from different websites that are reselling them, and put it in a dictionary with the websites as keys and the prices as the values. Assume that these websites are subject to change, and cannot be hardcoded. Write a function to help the TAs find the cheapest price available, returning the lowest price in the dictionary.

Note: We just want the lowest price, regardless of which site has it. If there is a tie for the lowest price, return any of the sites' lowest price.

Example Dictionary:

```
prices_dict = {"Vivid Seats": 150, "StubHub": 125, "SeatGeek": 149,
               "TicketMaster": 140, "AXS": 124}
```

Example return value: 124

```
def lowest_price(prices_dict):
    # Write your code below this line
    lowest_price = math.inf()
    for site in prices_dict.keys():
        if prices_dict[site] < lowest_price:
            lowest_price = prices_dict[site]
    return lowest_price
```


6. **[2 pts]** What is the content of the `animals.txt` file after the following code executes? Note: you may assume the file is empty before this code runs.

```
myfile = open("animals.txt", "w")

animals = ['wasp', 'butterfly', 'ant', 'moth', 'gecko', 'scorpion']

for animal in animals:
    myfile.write(animal[-1])

myfile.close()
```

python