CSE 160 - Midterm Exam Solution

Autumn 2023

Name: _____

UW Email: _____@uw.edu

- **Do not open this exam before the exam begins and close the booklet when time is called. Starting early or working after time is subject to a deduction.**

- You have **50 minutes** to complete this exam.

- This exam contains 8 pages (including this page) and 7 questions (6 exam questions, 1 extra-credit question).

- A Python syntax reference sheet is provided that includes documentation that we have covered in class.

- You are additionally allowed a 8.5" x 11" double-sided cheat sheet. All other materials besides writing utensils should be put away before the exam starts. This includes all devices like phones, calculators, and smartwatches.

- 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.

Question	Points	Score
1	4	
2	5	
3	5	
4	5	
5	5	
6	6	
Total:	30	

1. (a) (2 points) For each of the expressions below, write what 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). Remember to add quotation marks if the expression evaluates to a string.

   If an error is produced, write "Error" under the Evaluation column and leave the Type column blank.

Expression	Evaluation	Type
`int(3 + 1.5)`	4	int

`lst = [1, 3, 5, 7]`
 	
 `lst[-2] ** 2`
	25	int

`"NCT " + 127`	Error	
`"A" in "man" or 2 < 1`	False	bool

   (b) (2 points) Given the following dictionary, write what each expression evaluates to.

   ```
   random_dict = {"purple": 1, 2: "blue", "three": 3.0,

                  "hello": "world", 1: 100}
   ```

Expression	Evaluation	Type
`random_dict["hello"]`	"world"	string
`random_dict[100]`	Error	
`random_dict[purple]`	Error	
`random_dict["three"]`	3.0	float

2. Given the following function, what will print out when the below code is run?
   If nothing will be printed, write "Nothing".

```python
def mystery(n):
    if n < 0:
        print("Invalid input!")
    else:
        for i in range(1, n + 1):
            result = ""
            for j in range(i):
                result += "*"
            print(result)
```

(a) (1 point) `mystery(-1)`

Invalid input!

(b) (1 point) `mystery(0)`

Nothing

(c) (3 points) `mystery(5)`

```
*
**
***
****
*****
```

3. (5 points) You want to start a small flower garden at your house but don't know the right time to plant your seeds.

   Write a function `plant_time()` that takes in two parameters:

   - `season` - string which represents current season (can be only one of the values `"spring"`, `"summer"`, `"autumn"`, `"winter"`)
   - `temp` - integer which represents temperature in Fahrenheit

   Behavior:

   - If the season is `"spring"` or `"autumn"` you should return `"You can plant"` regardless of the temperature.
   - If the season is `"summer"` and the temperature is between 50 (inclusive) and 80 (inclusive), you should return `"You can plant"`, otherwise return `"Do not plant"`.
   - If the season is `"winter"` you should return `"Do not plant"` regardless of the temperature.

```python
def plant_time(season, temp):
    # Your code goes here

    if season == "spring" or season == "autumn":
        return "You can plant"
    elif season == "summer":
        if temperature >= 50 and temperature <= 80:
            return "You can plant"
        else:
            return "Do not plant"
    else:
        return "Do not plant"
```

4. (5 points) In each of the following code snippets, how many times is the line `"CSE 160"` (as one string on the same line) printed?

Code	# times

```for i in range(10):```
 	
```   print("CSE 160")```
	10

```for i in range(0, 9, 3):```
 	
```   print("CSE 160")```
	3

```for i in [1, 3, 5]:```
 	
```   print("CSE 160")```
 	
```   print("CSE 160")```
	6

```for i in range(5, 0):```
 	
```   print("CSE 160")```
	0

```for i in "hello world":```
 	
```   print("CSE 160")```
	11

5. (5 points) Given a list of strings that contains the names of TV shows `tv_shows` and a list of integers that contains the respective number of episodes per show `episodes`, write to a file `"shows.txt"` that contains a line `<show_name> has <episode_number> episodes` for each show.

   For example, given:

   ```
   tv_shows = ["Stranger Things", "Summer Strike", "Alice in Borderland"]
   episodes = [34, 12, 16]
   ```

   The file `shows.txt` will contain the content:

   ```
   Stranger Things has 34 episodes
   Summer Strike has 12 episodes
   Alice in Borderland has 16 episodes
   ```

   ```python
   f = open("shows.txt", "w")
   for i in range(len(tv_shows)):
       show = tv_shows[i]
       episode = episodes[i]
       f.write(show + " has " + str(episode) + " episodes\n")
   f.close()
   ```

6. Suppose we have the following list:

```
fruits = ["cherry", "strawberry", "elderberry", "raspberry"]
```

(a) (2 points) What is the printed output after running the following code?

```
ans = []
for i in range(len(fruits)):
    if i % 2 == 0:
        ans.append(fruits[i])
print(ans)
```

```
["cherry", "elderberry"]
```

(b) (2 points) What is the printed output after running the following code?

```
ans = []
for i in range(3):
    ans.append(fruits[i][0])
ans.extend(["160"])
print(ans)
```

```
["c", "s", "e", "160"]
```

(c) (2 points) Write code that modifies the original `fruits` list so that it contains the following:

```
fruits = ["cherry", "raspberry"]
```

```
There are a variety of answers to this problem. A few are listed below:

fruits.remove("strawberry")
fruits.remove("elderberry")

OR

fruits.pop(1)
fruits.pop(1)

OR

fruits[1:3] = []
```

7. (1 point) Extra Credit: Please write down two names of the course staff! Just the first name is fine.

Andre
Aneesha
Annalisa
Aparna
Diana
Hannah
Jason
Lucas
Sebastin
Sheamin
Sierrah
Suh Young
Suhas
Therese
Vatsal