

Name:

Email ID:

On my honor, I pledge that I have neither given nor received help on this test.

Signature:

Testing

- Print your name, id, and pledge as requested.
- This pledged exam is closed textbook. The only device you may access during the test is your own laptop.
- The only web material you may access during the test are class web pages.
- You are not allowed to access class examples or your own past assignments during the test; i.e., the only Python code you may access or view are ones that you develop for this test.
- You are not allowed to use PyCharm for the pen and paper part of the test.
- The only windows that can be open on your computer are PyCharm and a single browser with tabs only open to the class website.

Short answer (20 points)

1. What type of value does the `input()` function return?

2. Suppose `s` is a string. How can you determine its length?

3. What is the difference between the `/` and `//` operators?

4. What statement is necessary in order to get program access to the trigonometric functions `sin()`, `cos()`, and `tan()`?

5. What does the expression `(a % b)` calculate?

6. Suppose `s` is the string `'abc'`. What is the value of `s[1]`?

7. Is `n` the last sequence value produced with the `range(1, n)`?

8. Suppose `x` is the list `[3, 1, 4]`. What type of value does `x[1 : 2]` return?

9. What is the value of the expression `(1 // 2 * 2)`?

10. Why isn't Python able to represent all decimals perfectly?

True or False (5 points)

11. True False `(3 + 4.0)` is a legal Python expression.

12. True False `(3 + '4.0')` is a legal Python expression.

13. True False The value of the expression `(True or False)` is False.

14. True False `sum + 1 = sum` increments `sum` by 1.

15. True False Suppose `s` is a string. The invocation `s.lower()` changes `s` so that all of its letters are now lowercase.

Problem solving

16. (15 points) Develop program `success.py`

- The program prints the string *success* with all letters in lowercase except for the initial *s*. There should be no other output.
- A program run

```
Success
```

17. (20 points) Develop program `fore.py`

- The program prompts and gets four strings from its user. The program prints those strings in sorted order one per line. There should be no other output.
- Two program runs

```
Enter four strings: s a aa b
a
aa
b
s
```

```
Enter four strings: a1 1a b 1
1
1a
a1
b
```

18. (20 points) Develop program `octane.py`

- The program prompts and gets three integers s , n , and d from its user (in that order). The program uses integer s as a seed to the Python random number generator. The program first prints n octal digits (base 8 digits) line by line. It then prints the number of generated occurrences of d . There should be no other output. Hint: a list might prove helpful.
- Two program runs (your program should produce the same outputs when run with the indicated inputs).

```
Enter three numbers: 7 8 1
5
2
6
0
1
1
5
0
2
```

```
Enter three numbers: 11 12 2
7
7
7
3
2
7
2
1
7
4
2
1
3
```

19. (20 points) Develop program `elevate.py`

- The program reads the dataset at `http://www.cs.virginia.edu/~cs1112/datasets/csv/land-mass.csv` where
 - The first row in the dataset is a header.
 - Each row in the dataset has three columns with
 - The first column being the name of a land mass;
 - The second column being the maximum elevation in meters for that land mass; and
 - The third column is the minimum elevation in meters for that land mass.
- The current contents of the dataset are given below. The contents will be different during grading.

```
Land mass,Max elevation (meters),Min elevation (meters)
Narnia,4810,-10
Neverland,426,-2
Oz,1231,679
Sleepy Hollow,1629,304
Stars Hollow,725,152
Toyland,6187,0
Wonderland,5895,-5
```

- The program prints on separate lines, the max elevation for the land masses, the min elevation for the land masses, and the difference of those two elevations. There should be no other output.
- Program run with the above contents for `land-mass.csv`

```
6187
-10
6197
```