

Ever so clearly print your computing id:

Ever so clearly print your name:

Pledge:

Test rules

- Before you leave the room, check that you uploaded all of your solutions. Do not ask afterwards whether you can submit a forgotten solution.
- This pledged exam is closed notes. The only device you may access during the test is your laptop.
- Any cheating can result in failing the class and the incident being referred to the Honor Committee.
- Do not access class examples artifacts, web solutions, or your own past assignments during the test; that is, the only code you may access or view are ones that you develop for this test.
- The only windows allowed on your laptop are PyCharm and a single browser with tabs reachable from class website.
- The Part 1 problems are each worth two points; the Part 2 problems are each worth twelve points.

PyCharm

- PyCharm can be used for developing the modules to be submitted. It **cannot be used** for the short answer questions of Part 1.

Program

- Modules should follow class programming practices; e.g., whitespace, identifier naming, and commenting if you think it is needed, etc.
- Whether a program runs is important.
- Comment out or delete all debugging `print()` statements before submitting.

Short answer questions answers

1. _____

8. _____

2. _____

9. _____

3. _____

10. _____

4. _____

11. _____

5. _____

12. _____

6. _____

13. _____

7. _____

14. _____

THIS PAGE IS ALMOST BLANK

Part 1: short answer questions

1. TRUE or FALSE In an assignment statement, the variable to be updated appears to the left of the = operator.

2. TRUE or FALSE In the following code segment, 10 is the only variable.

```
x = 10 // 3
```

3. TRUE or FALSE Only programs with snake_case variables can run correctly.

4. What built-in function converts a string into an integer?

5. What built-in function tells you the length of a string?

6. What built-in function tells you the length of a list?

7. What does the following code segment print?

```
x = 10 // 3
print( x )
```

8. What does the following code segment print?

```
x = 3 * 10 // 3
print( x )
```

9. What does the following code segment print?

```
x = 2019 % 100
print( x )
```

10. What does the following code segment print?

```
s = "every"
character = s[ 1 ]
print( character )
```

11. What does the following code segment print?

```
s = "every"
f = s.find( "e" )
print( f )
```

12. What does the following code segment print?

```
EMPTY_STRING = ""
s = "every"
r = s.replace( "e", EMPTY_STRING )
print( r )
```

13. How many times does the string "again" get printed in the following code segment?

```
for r in range( 0, 4 ) :
    print( "again" )
```

14. How many times does the string "again" get printed in the following code segment?

```
for r in range( 0, 4 ) :
    for c in range( 0, 3 ) :
        print( "again" )
```

Part 2: Programming

15. Complete program *honorable.py* whose only output is the string "I am honorable." The program does not get any user input. A run of the program should always produce the following.

```
I am honorable.
```

16. Complete program *future.py* that prompts for the age of its user. The program computes and prints how old a person of that age would be in the year 2100. There should be no other output. The program defines two constants. They are to be used in your computation.

```
FUTURE_YEAR_OF_INTEREST = 2100
CURRENT_YEAR_OF_INTEREST = 2019
```

Three possible program runs follow.

```
Enter your age: 19
100
```

```
Enter your age: 32
113
```

```
Enter your age: 46
127
```

17. Complete program *gimme.py*. The program separately prompts for two lines of the text. The program computes the list of words gotten by combining the words from the first line of text followed by the words of the second line of text.

The program outputs on separate lines: the words in the combined list, the word in the combined list that occurs first alphabetically, and the word that occurs last alphabetically. There should be no other output. Two possible program runs follow.

```
Enter text: a b c
Enter words: d e f g h i j k
['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k']
a
k
```

```
Enter text: love is all you need
Enter words: long long ago
['love', 'is', 'all', 'you', 'need', 'long', 'long', 'ago']
ago
you
```

18. Complete program *worthy.py*. The program has one prompt to get a line of text. The program computes two lists: the list of words supplied in the user text and a list of the lengths of those words. The program outputs on separate lines: the list of words and the list of word lengths. There should be *no other output*. Two possible program runs follow.

```
Enter text: You will do good on this test
['You', 'will', 'do', 'good', 'on', 'this', 'test']
[3, 4, 2, 4, 2, 4, 4]
```

```
Enter text: I know I will
['I', 'know', 'I', 'will']
[1, 4, 1, 4]
```

19. Complete program *huh.py*. The program separately prompts to gets three inputs from the user. The first input is a word; the second input is text; the third input is an integer.
- The first input is to be used as a seed value for the Python random number generator.
 - The second input is to be split into a list of words.
 - The third input determines the number of random choices made on those words.

For the indicated number of selections, the program randomly picks one of the input words and prints it on its own line. There should be *no other output*. Two possible program runs follow.

```
Enter seed: seed
Enter text: b f a c e d g h
Enter sentence length: 5
a
b
c
d
e
```

```
Enter seed: mlk
Enter text: a are have hope I dream good hello bye
Enter sentence length: 4
I
have
a
dream
```

20. (Complete program *in_all.py*. The program *does not get* any user input. The program analyzes the web file at location.

<http://www.cs.virginia.edu/~cs1112/numbers.txt>

The web file is a list of integers; that is, it is not a CSV file. While you are taking the exam, the file contents are

```
31
41 59 26 53 58 97 93 23 84 62 64 33 83
279 502 884 197 169 399 375 105
82 89 74 94 45 92 30 78 16 40 62
```

The program prints the maximum integer in the data file. During the exam, the output of the program should be

```
884
```

Be aware that both the number of lines and contents of the data file will change during the grading of the exam.