CSC 2280
Introduction to Computer Science
Florida Southern College

Assignment 3
Due: Friday, March 22, 2018

True/False (24 points)

1. False A Python string is always enclosed in double quotes.
2. False The 2nd to last character of a string `s` (e.g. `'a'` in `'Friday'`) is at position `len(s)-1`.
3. False The `add` method can be used to add an item to the end of a list.
4. **True** The `split` method breaks a string into a list of substrings, and `join` does the opposite.
5. True Python lists are mutable, but strings are not.
6. False In Python, a function can return only one value.
7. True Variables defined in a function are local to that function.
8. False It is a bad idea to define new functions if it makes a program longer.

Multiple Choice (24 points)

9. B If `s="Strings are fun!"`, which of the following is the same as `s[2:]`?
   (a) `s[:2]` (b) `s[2:len(s)]`
   (c) `s[2:len(s)-1]` (d) `s[:]`
10. D The string "slots" that are filled in by the format method are marked by:
   (a) `%` (b) `[]`
   (c) `$` (d) `{}`
11. A Which of the following is *not* a file-reading method for Python?
   (a) `readall` (b) `read`
   (c) `readline` (d) `readlines`
12. C Accessing a single character out of a string is called:
   (a) slicing (b) concatenation
   (c) indexing (d) assignment
13. C In Python, `"4"+"5"` is:
   (a) `9` (b) `"9"`
   (c) `"45"` (d) `45`

14. D Which string method converts all the characters of a string to upper case?
(a) `capitalize` (b) `title`
(c) `uppercase` (d) `upper`

15. C A function can send output back to the program with a(n)
(a) `print` (b) `assignment`
(c) `return` (d) `output`

16. B In Python, actual parameters are passed to functions
(a) by value (b) by reference
(c) at random (d) by networking

Short Answer (20 points)

What is the output for each of the following code fragments in Python? If the code produces an error, explain why. (Be careful about `float` vs `int`!)

17. 31
```
7 + 6 * (5 - 43 % 21)
```

18. 2
```
28 // 5 - 28 % 5
```

19. 8.0
```
5 + 2 / 4 * 6
```

20. 1861
```
int((61**2 + 1860**2)**(1/2))
```

21. 42.0
```
float(int((abs(int(-4.5)) + 0.25) * 5) * 2)
```

22.
```
x = "roads"
quote = "{1}? Where we're going, we don't need {0}."
quote.format(x, x.capitalize())
```

‘Roads? Where we’re going, we don’t need roads.’

23.
```
"Hello, {} {} {}".format("Dr.", "Sheldon", "Cooper")
```

‘Hello, Dr. Sheldon Cooper’

24.
```
"{:0.3f} {:0.3f}".format(3.14, 3.14159)
```

‘3.140 3.142

25.
```
"My name is {1}, I am a {2}, I am taking {3} credit hours, and {4} is
my favorite class".format("Howard", "freshman", 18, "CSC 2280")
```

It produces an error that reads “tuple index out of range” because Python indexes from zero and the formatting parameters start at 1

26.
```
for w in "Alabama".split("a"):
    print(w, end=" ")
```

    Al b m

Short Programming (15 points)

27. Given the list, `words = ["rock", "paper", "scissors", ""]`, write one line of Python code that would return the following:
`'Rock!Paper!Scissors!'`

```
"!".join(words).title()
```

28. Given a sentence, write one line of Python code that returns the number of words in a sentence. For example, if `sentence = "I'm a Ramblin' Wreck from Georgia Tech, and a hell of an engineer!"`, then your code should return 13.

```
sentence.count(" ") + 1
```

29. Using the strings,

```
str1 = "??????????python programming"
str2 = "====: ===="
str3 = "An Introduction to Computer Science"
str4 = " by John Zelle!!!!!!!!!!!!!!!!!!!"
```

    write one line of Python code that returns the following string:

    `'Python Programming: An Introduction to Computer Science by John Zelle'`

```
print(str1[10:].title(),str2[4], str3, str4[1:13])
```

30. Given `quote = "The University of Alabama is the best football team in Alabama."`, write one line of Python code that replaces `Alabama` with `Florida` and assigns it to the variable `new_quote`. (You must use the variable `quote` in your code!)

```
new_quote = quote.replace("Alabama", "Florida")
```

Programming (20 points)

31. *(10 points)* A Caesar cipher is a simple substitution cipher based on the idea of shifting each letter of a plaintext message by a fixed number (called the `key`) of positions in the alphabet. For example, if the `key` is 2, the word "secret" would be encoded as "ugetgv" (s→u, e→g, c→e, ...). The original message can be recovered by decoding it using the negative of the key.

    For this problem, your task is to write a Python function that implements a Caesar cipher.
    - The function must take two inputs: a string and a key
    - The function must return one output: the encoded string

    The function has been started for you in the file cipher.py. Modify this code as indicated and include the file in your homework submission.

    NOTE: For reference, the required modifications can be done in as little as 4 lines of code.

    Using your function, what is the encoded version of the string `"weekend"` if `key=1`?

    x f f l f o e

    BONUS: *(10 points max)* You may have noticed that the cipher above converts letters near the end of the alphabet to non-letters. For example, `cipher("xyz",3) = "{|}"`. A true Caesar cipher does the shifting in a circular fashion where the next character after `"z"` is `"a"`. Modify your cipher code to make it circular. You may assume the input string will only consist of letters.

32. *(10 points)* Many word processing programs allow the user to view properties of the text contained therein. For this problem, your task is to write a Python function that reads a text file and computes the total number of words and characters in the file.

    The function has been started for you in the file wordcount.py. Modify this code as indicated and include the file in your homework submission.

    Print (do not return any outputs) the statistics so that is looks identical to the sample below:

    ```
    Processing file: sample.txt
    Characters: 445
    Words: 69
    ```

    NOTE: For reference, the required modifications can be done in as little as 8-10 lines of code.

    USE <return txt> not <print txt>