CMPS 5P
Introduction to Programming in Python
Final Exam Review Problems

1. What is printed by the following Python program?

```python
name = input('Please enter a sentence: ')
print('Your sentence has',len(name),'characters')
```

Assume the user enters the following text: The cow jumped over the moon

2. Write a complete Python program that prompts for and gets the name of a file in the current working directory, opens that file for reading, then prints out the file line by line in reverse order, i.e. last line first and first line last.

3. This is exercise 5 in section 2.14 of the online text. The formula for computing the final amount if one is earning compound interest is given on Wikipedia as

\[ A = P \left(1 + \frac{r}{n}\right)^{nt} \]

Write a Python program that assigns the principal amount of $10,000 to variable \( P \), assign to \( n \) the value 12, and assign to \( r \) the interest rate of 8%. (Note \( r \) must be expressed as a number, not a percent, so \( r = .08 \) not 8.) Then have the program prompt the user for the number of years \( t \) that the money will be compounded for. Calculate and print the final amount after \( t \) years.

Re-do this problem by writing a function \( \text{Amount}(P, r, n, t) \) that calculates the above formula.

4. What is printed by the following Python program?

```python
def foo(a, b):
    print('in foo()')
    return a+b
# end foo

def bar(a, b):
    print('in bar()')
    return a*b
# end bar

# -- main-----------------------------------------------
x = 'aa'
y = 'bbb'
z = 4
s = 'foo({0},{1})={2}'.format(x,y,foo(x,y))
t = 'bar({0},{1})={2}'.format(z,y,bar(z,y))
u = 'foo(bar({0},{1}),{2})={3}'.format(z,x,y,foo(bar(z,x),y))
print(s)
print(t)
```
5. Write a Python function `sum_cubes(n)` that returns the sum of the cubes of the integers from 1 to `n`. Thus `sum_cubes(5)` would return $1^3 + 2^3 + 3^3 + 4^3 + 5^3 = 225$. Use a loop to calculate this sum, not a formula that you may know. Generalize this function by writing `sum_powers(n, k)` that returns the sum of the $k^{th}$ powers of the integers from 1 to $n$. Use this function to determine the number of digits in `sum_powers(100, 100)`. (Answer: 201 digits).

6. Write a Python function called `swap(L, i, j)` that takes as input a list $L$ and two ints $i$ and $j$, then returns a new list that is a copy of $L$, except that the elements at indices $i$ and $j$ are exchanged.

7. What is printed by the following Python program?

```python
L = ['123', '132', '213', '231', '312', '321']
S = ['one', 'two', 'three']
for p in L:
    x = S[int(p[0]) - 1]
    y = S[int(p[1]) - 1]
    z = S[int(p[2]) - 1]
    print(x, y, z)
# end for
```

8. What is printed by the following Python program?

```python
x = 'happy'
y = 'sad'
if x < y:
    for i in range(1, 10):
        print(10 - 2 * i, end=' ')  
else:
    i = 1
    while i < 10:
        print(3 * i, end=' ')
        i += 1
    print()
print('i =', i)
```

9. Write a Python function called `area_of(s, x)` that returns the area of a circle of radius $x$ if $s$ is the string 'circle', returns the area of a square of side $x$ if $s$ is the string 'square', and returns None if $s$ is anything other than one of those two strings.

10. Write a Python function called `notDivisibleBy(a, n)` that returns a list of all integers in the range 1 to $n$ (inclusive) that are not divisible by $a$. For instance, the call

```python
notDivisibleBy(4, 12)
```

will return the list:

```
[1, 2, 3, 5, 6, 7, 9, 10, 11]
```
11. Write a Python function called `print_rectangle(n, m)` that prints out a rectangle made of '***' characters with $n$ rows and $m$ columns. For instance the call `print_rectangle(4, 18)` would print the following picture.

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

Modify your solution to this problem by writing a function called `print_open_rectangle(n, m)` that prints a rectangle of the same size, but made of '***' characters on the boundary, and filled with spaces in the interior. Thus the call `print_open_rectangle(4, 18)` would print the following picture.

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

12. A Palindrome is a word (or string) that spells the same forward as backward. For example ‘kayak’ and ‘straw warts’ are Palindromes while ‘horse’ and ‘market’ are not Palindromes. Write a Python function called `isPalindrome(s)` that returns `True` if its string argument $s$ is a Palindrome, and returns `False` otherwise.

13. Write a function called `remove_from(s, i, j)` that returns a new string obtained from string $s$ by removing the characters from index $i$ (inclusive) to index $j$ (exclusive). For instance, the call `remove_from('abcdefg', 2, 5)` will return the string 'abfgh'. (Hint: you can do this in a single line by using slices). Write another function called `remove_sub(s, t)` that returns a new string obtained from string $s$ by removing the leftmost occurrence of the substring $t$, if $t$ is a substring of $s$. If $t$ is not a substring of $s$, return $s$ unchanged. Thus `remove_sub('xxxabcyyyyabcczzz', 'abc')` returns the string 'xxxyyyyabcczzz'. (Hint: use the built-in string function `find()`, which is explained in section 4.7.1 of the Python Library Reference at https://docs.python.org/3/library/index.html.)

14. Determine the output of the following Python program.

```python
def cleanUp(L):
    y = L[0]
    M = [y]
    for x in L[1:]:
        if x != y:
            y = x
            M.append(y)
    return M

#-- main --------------------------
A = [1,5,3,7,3,4,3,5,4,7]
A.sort()
B=cleanUp(A)
for x in B:
    print(x)
```

15. Complete the definition of the module `average.py` below. The two functions `avg_list()` and `avg_nested_list()` are described in the corresponding doc strings.

```python
# average.py

def avg_list(L):
    """
    Returns the average of all the numbers in list L, assuming L contains only numbers.
    """

# end avg_list()

def avg_nested_list(L):
    """
    Returns the average of all the numbers in all the elements of the list L, assuming that each element of L is a list containing only numbers. Note that L may be ragged, i.e. not all elements of L need have the same length.
    """

# end avg_nested_list()

Test your module by running the following commands (either in interactive mode, or in a Python program.)

```python
import average
A = [1, 2, 3]
B = [4, 5]
C = [6, 7, 8, 9]
print(average.avg_list(A))  # prints 2.0
print(average.avg_nested_list([A, B, C]))  # prints 5.0
```