Chapter 4 Exercises

Instructions: Complete the following exercises using Python 3. You may work with up to one other person. You may use library packages unless you are explicitly told not to. If a problem tells you to write a program/function for arbitrary input, you should always test your function and ensure it works. It is critical that you complete and understand *every* problem on this assignment. If you are stuck or need any help, please do not hesitate to ask the instructor or teaching assistant.

1) Write a program that adds the irrational numbers e and π. Print the result to console to 4 decimal places.

2) Write a program that calculate the sum of the first n positive integers, where n is any user-defined integer greater than zero.

3) Write a program that appends n random integers to a list, where n is any number greater than zero. Determine if each number in the list is even or odd. If it is even, print both the number and a statement saying it is even to console. If the number is odd, print both the number and a statement saying it is odd to console.

4) You may not use library functions for this problem. Create two lists. In one list, store the first n even numbers. In the other list, store the first n odd numbers. Here, n is any integer greater than zero.

5) The Fibonacci sequence is obtained by starting with the sequence 1, 1, then recursively adding the last two terms together to generate the next term. The first few terms of the Fibonacci sequence are 1, 1, 2, 3, 5, 8. Write a program that calculates the Fibonacci sequence to n terms. What is the $n-th$ term of the Fibonacci sequence?

6) Write a program that returns the data type of the following items: π, e, 3, 3.0, $'3.0'$, $3+2i$ (i is $\sqrt{-1}$), $True$, 2.54, $[1,3,4,5]$, $(3,5,6)$, $[[1,2],[3,4]]$.

7) Write a function that calculates the scalar product between two vectors. Write a separate function that calculates the angle between two vectors. Check your answer for arbitrary vectors.

8) Write a function that returns the cross-product of two vectors.

9) Write a function that returns the surface area and volume of an arbitrary sphere of radius r in units of meters and inches. Print your results to console.

10) Three satellites are in orbit around Earth. Their orbital periods are 45 minutes, 90 minutes, and 160 minutes, respectively. Write a function that determines the orbital altitude of each satellite above Earth's surface. Comment on your results. (Hint: Use Kepler's third law.)

11) Write a program that generates a list of n random numbers, where n is some integer greater than zero. For the next parts, you may not use library functions. Write a function that determines the largest number in the list. Reverse the order of the list. Determine if an arbitrary number is in the list.

12) Write a program that generates a 10×10 matrix whose elements are random integers between 0 and 9. For the following parts, you may not use library functions (except np.array()). Write a function that returns the transpose of the matrix you generated. In your transposed matrix, swap the first and last rows with each other. Next, swap the fist and last columns with each other. Now, replace all $8's$ in your matrix with the integer 10. What is the resultant matrix?

13) Write a program that creates a list of n zeros, where n is some integer greater than zero. Replace every second zero with the string 'foo'. Replace all remaining zeros except the first with random integer multiples of π.

14) You may not use library functions. For integers numbers between 1 and 100, if a number is divisible by 3 print the string 'fizz' to the console. If a number is divisible by 5, print the string 'buzz' to the console. If the number is divisible by both 3 and 5, return the string 'fizz buzz' to the console. Otherwise, return the number itself.

15) You may not use library functions. Write a function that determines if a number is prime. If the number is prime, return $True$ and print the number to the console, and say it is prime. If the number is not prime, return $False$ and print the number to the console saying it is not prime, and state the divisors of the number.

16) You may not use library functions. Write a function that returns the factorial of a positive integer.

17) You may not use library functions. Write a function called 'sin' that calculates the the Taylor series expansion of $\sin(x)$ about zero to the $n - th$ term. Do the same thing for cosine(x).

18) Generate a .csv file (a file whose entries are separated by commas) that contains the first 500 prime numbers in a 'tree' format. In other words, the beginning of your file should

look like this:

```
2
2,3
2,3,5
2,3,5,7
```

Hint: Use the function you wrote in problem (15).

19) Write a program that reads the .csv file you generated in Exercise (18). Take the last line of that .csv file and put it into a tuple. Write a function that computes the average and standard deviation of all of the numbers in that tuple.

20) Write a program that plots the following functions: x^2, $x^3 + 2x + 4$, $\sin(x)$, $ln(x)$, e^x, 3^x, $sin(x)/x$. Plot all polynomials on one graph, all trig functions on a separate graph, and the natural log and exponentials on a third graph. Use bounds you feel are appropriate. Make sure your graphs are readable (make the functions in the same graphs different colors or have different line styles), and include a legend, grid, title, and axes labels.

21) Plot x^2, $\sin^2(x)$ and $\sin(x^2)$ on separate graphs using 10, 25 , 50,100, and 1000 data points. You should have 3 graphs, each graph having 5 functions of varying quality (based on the number of data points you use). Include a legend, grid, title, and axes labels in your plots.

22) Write a function that returns the distance between two points in 3-dimensional space.

23) A p-norm is defined as

$$||x_p|| \equiv \left(\sum_i^n |x_i|^p \right)^{1/p}$$
$$||x_\infty|| \equiv max(x_i)$$

for a collection of n numbers. Here, p can be any integer. For $x = [2, 6, 4, -7, 18, 34, -3, 9, 6]$, compute the p-1, p-2, and p-∞ norms.

24) Generate a list of 1,000 random numbers between 0 and 100. Plot a histogram of the distribution. Do the same thing again, except draw the numbers randomly from a Gaussian distribution.

25) Plot a circle of radius r. Include a title, legend, grid, and axes labels in your plot.

26) Generate a random list of 100 positive integers. You may not use library functions for the next part. Remove all duplicate numbers from the list.