1. Prompt the user for a number. Compute twice that number, half that number, add 50 to it, and subtract 50 from it. Print those results
2. Prompt the user for 4 numbers
3. Store them all in a list
4. Compute the sum of all those numbers
5. Make a new list that is the product of each pair of elements in the original list (e.g. if the original list is [3,6,8,2], then the new list should be [18,16])
6. Print all these things out
7. Prompt the user for the names of 3 friends and their ages
8. Store this in a dictionary
9. Print the name of each friend and how old they are now, as well as how old they’ll be in 5 years
10. Let’s break things
11. What happens if you compute 5/7? Can you figure out why? Can you figure out how to fix it?
12. Make a list of length 5. Try to access element 5. What happens? Why?
13. Make a dictionary. Try to access an element that doesn’t exist.
14. Make a dictionary. Try to use a list as a key. What happens?
15. Let’s use a loop to make it easier to prompt the user for numbers
16. Redo problem 2, but use a for loop
17. Redo problem 2, but allow the user to specify an arbitrarily long sequence of numbers (hint: you might ask the user to input a special character if they want to stop inputting characters)
18. Prompt the user for numbers until they want to stop (see 5b). For each number, if the number is EVEN, tell the user the number is even and print half that number. If the number is ODD, tell the user the number is odd, and print twice the number. Hint: to check for evenness and oddness, the “modulo” (or remainder) operator is probably useful.
19. Write a function that takes a list of numbers as an argument and returns the product of the first and last elements of that list. Use it in a script that prompts the user for 5 numbers to return the product of the first and last numbers the user input.
20. Super hard final problem: on the blackboard, there is a file called numbers.txt. It is a tab separated file where each line contains some numbers.
21. Read the file line by line, turning each line into a list of numbers
22. For each line, use a function to compute the minimum and maximum elements of that line (hint: you can return multiple items from a one function!)
23. Write the output to a file that for every line says what the minimum and maximum of those lines are. Use string substitution to do this in complete sentences.