1. (a) Start Chrome browser.
   (b) Go to http://cocalc.com and “Sign In”.
   (c) Click project Math 356.
   (d) Click “New”, call it c08, then click “Sage Worksheet”.

2. Lists can be combined with “+”. Evaluate [2,5,9]+[3,4,5].

3. If you want all the integers from \(x\) to \(y\) you can use the shorthand notation \([x..y]\). Evaluate [3..7].

4. If you want a list with \(m\) \(n\)’s you can use the shorthand notation \([n]*m\). Evaluate [0]*7.

5. You can have a list of lists. Evaluate L=[[0,1],[2,3],[4,5]]. Now evaluate L[1]. Then evaluate L[1][0]. What do you think the value of L[0][1] is?

6. You can use map() to apply a function to each term of a list. Evaluate map(abs,[-1,2,-3]).

7. Use list comprehension to produce a list of the cubes of all the integers from 2 to 17.

8. List comprehension can also be used to filter the numbers in a list. Evaluate \([x \text{ for } x \text{ in } [2,5,9] \text{ if } x%2==0]\). What did this do?

9. Evaluate \([x \text{ for } x \text{ in } [2,5,9] \text{ if } x%2==1]\). What did this do?

10. Let L=[5,7,9,11,13]. Find the length of \(L\) by evaluating len(L).

11. You can add an element to the end of list with append. Evaluate L.append(47). Now evaluate L.

12. You can pop an item off of the end of a list. Try L.pop(). Now evaluate L. Repeat. Try L.pop() again. And then evaluate L again.

13. You can reverse a list. Try L.reverse(). Now evaluate L.
14. You can **sort** a list (from smallest to biggest). Let \( M=[3,2,6,5,3,4,9,1] \). Try \( M.sort() \). Now evaluate \( M \).

15. You can **count** the number of \( x \)'s in a list \( M \) with \( M.count(x) \). Try \( M.count(3) \).

16. Evaluate the following function definition:

   ```python
   def hellos():
       for i in [1,2,3]:
           print "hello!"
   
   Now evaluate hellos()
   ```

17. Evaluate the following function definition:

   ```python
   def hellos2():
       for i in [4,5,6]:
           print "hello!"
   
   Now evaluate hellos2(). Is there any difference. Why?
   ```

18. Evaluate the following function definition:

   ```python
   def hellos3(n):
       for i in [1..n]:
           print "hello!"
   
   Now evaluate hellos3(5) and hellos3(7)
   ```

19. Define a function that takes each element from a list, doubles it, and returns a new list with the doubled elements:

   ```python
   def double(L):
       M=[]
       for x in L:
           M.append(2*x)
       return M
   
   Now try double([2,3]).
   ```

20. Input the code from p.14 of our book (p.20 of the pdf). See if you can start to figure out what it **means**.

21. Seeing the graph might help. Try \( g.show() \).

22. You can often get some explanation for what a command does by typing a question mark after the command name. Try \( g.shortest\_path? \)