1. What exactly is []?

**Empty list**

2. In a list of values stored in a variable called spam, how would you assign the value 'hello' as the third value? (Assume [2, 4, 6, 8, 10] are in spam.)

**Spam[2] = ‘hello’**

Let's pretend the spam includes the list ['a', 'b', 'c', 'd'] for the next three queries.

3. What is the value of spam[int(int('3' \* 2) / 11)]?

**d**

4. What is the value of spam[-1]?

**d**

5. What is the value of spam[:2]?

**[‘a’,’b’]**

Let's pretend bacon has the list **[3.14, 'cat,' 11, 'cat,' True]** for the next three questions.

Assuming there was error and the correct list is **bacon=[3.14, 'cat', 11, 'cat', True]**

6. What is the value of bacon.index('cat')?

**1**

7. How does bacon.append(99) change the look of the list value in bacon?

**[3.14, 'cat', 11, 'cat', True,99]**

8. How does bacon.remove('cat') change the look of the list in bacon?

**[3.14, 11, 'cat', True, 99]**

9. What are the list concatenation and list replication operators?

**+,\* respectively**

10. What is difference between the list methods append() and insert()?

**Append() – Adds to the end of the list**

**Insert() – It can add anywhere doesn’t follow any order**

11. What are the two methods for removing items from a list?

**Remove()**

**del**

12. Describe how list values and string values are identical.

**Both are sequential collections.**

**Indexes, Len(), Can be used in loops, operators in/Not in**

13. What's the difference between tuples and lists?

**Lists are enclosed in [] and are mutable**

**Tuples are enclosed in () and are immutable**

14. How do you type a tuple value that only contains the integer 42?

**(42,)**

15. How do you get a list value's tuple form? How do you get a tuple value's list form?

**Tuple() and list()**

16. Variables that "contain" list values are not necessarily lists themselves. Instead, what do they contain?

**Contains references to list values**

17. How do you distinguish between copy.copy() and copy.deepcopy**()?**

**Copy.copy() – Shallow copy constructs a new compound and then inserts references into it from the original.**

**Copy.deepcopy() -- constructs a new compound and then recursively inserts references into it from the original.**