1. What exactly is []?

---place holder use to create 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.insert(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.

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]

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

---concatenation: +, extend ! replication: \*

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

---append add object\element at the last index of list while insert used for adding element\object at particular given index

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

---pop() and remove()

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

---both are sequence datatype, hence iterables

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

---tuples are immutable while lists are mutable

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

--- a = tuple('42') ! b = (‘42’,)

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

---using zip() , iter(), map() function and tuple()

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

---it may contain any datatype like string, int, float, Boolean, etc

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

---copy() creates reference to original object while deepcopy() create real copy of original object and hence new object using deepcopy()