**1. What exactly is []?**

[] 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.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, 'cat,' 11, True]

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

+ is concatenation and \* is replication operators

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

Append will add element at the end while insert will add element at specified index

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

Pop can remove element by using index no. and remove will remove element by specifying its valve

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

Both are ordered collections of characters. E.g. We can access second element in list by list[1] or of string by string[1]

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

Values inside tuples can not be changed once it is stored while value inside lists can be changed

**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?**

Suppose l=[1,2,3] and t=(4,5,6)

From list to tuple: tuple(l)

From tuple to list:list(t)

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

They contain reference to list value instead of list itself.

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

copy() create reference to original object. If we change copied object - you change the original object. .deepcopy() creates new object and does real copying of original object to new one. Changing new deepcopied object doesn't affect original object.