1. What exactly is []?

Ans: [] is an 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.)

Ans: 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)]?

Ans: ‘d’

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

Ans: ‘d’

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

Ans: [‘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')?

Ans: 1

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

Ans: Appends 99 to the list bacon.

ie; [3.14, 'cat,' 11, 'cat,' True,99]

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

Ans: Removes the first occurrence of ‘cat’.

ie; [3.14, 11, 'cat', True, 99]

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

Ans: The operator for list concatenation is +, and the operator for replication is \*.

Example:  
 >>> A = [1,2,3]  
 >>> B = [4,5,6]  
 >>> C = [1,2,3] + [4,5,6]  
 >>> C  
 [1,2,3] + [4,5,6]   
 >>> R = [1,2,3]\*3  
 >>> R  
 [1, 2, 3, 1, 2, 3, 1, 2, 3]

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

Ans: append() adds an element at the end of the list. Syntax: list\_name.append(element)

insert() can be used to add element at any position of the list.   
 Syntax: list\_name.append(index, element)

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

Ans: They are:  
1. remove() : Removes specified element or item from the list.   
Syntax: list\_name.remove(element)

2. pop() : Removes element or item at given position or index of the list and returns the element.   
Syntax: list\_name.pop(index)  
If no index is specified then it removes the last element.

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

Ans: They both are sequences. String is a sequence of characters whereas list is sequence of item of different data types. Thus, they both have length and index, and can be used similarly in for loop, slicing, concatenation and replication operations.

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

Ans: Lists are mutable, ie; items in the list can be added, removed, or changed. Tuples are immutable, i.e; they cannot be changed. Items in the list are written inside square brackets [ ], whereas items in the tuple are written inside parentheses ( ).

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

Ans: (42)

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

Ans: Using functions list() and tuple() respectively.

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

Ans: They contain references to list values.

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

Ans: copy.copy() is a shallow copy. It doesn't create a copy of nested objects, instead it just copies the reference of nested objects. Changing the values of nested objects will also change the values of nested objects in original object.

copy.deepcopy() is a deep copy. The deep copy creates independent copy of original object and all its nested objects. Changing the values of nested objects will not change the values of nested objects in original object.