1. What exactly is []?
   Ans: it's 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, 4, 6, 8, 10]

   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: [3.14,"cat",11,"cat",True,99]
```

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

Ans: [3.14,11,"cat",True]

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

Ans: For the list concatenation "+" and for replication "*"

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

Ans: append() adds the data at the end of the list

   insert() insert an item at specified index

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

Ans: remove(), pop()

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

Ans: List values and string values both are used to store value/ data and both are sequences

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

Ans:Tuple are immutable while list are mutable

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

Ans: t=(42)

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

Ans:l=[1,2,3]   t=tuple(l) -list to tuple

   t=(1,2,3)   l=list(t)=tuple to list

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

Ans: List is value that contains multiple values in an ordered sequence.

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

Ans: copy.copy() returns shallow copy of list while deepcopy() returns deepcopy of list