3-14

a) The for loop runs **N** times,

  - **N** = the length of the List L.
- So the Time Complexity of the mean(L) function is **O(N).**
- This means it is a linear time complexity.

Proof:

- Worst Case: total number of times the for loops gets executed will be N times
  - **N** = length of L
- The operation of (total += x) are executed at constant time.
- The return values are also executed at constant time.
- Therefore, the time complexity of the mean(L) function is **O(N)**.

b) The first loop gets executed **n** times

  - **n** is a positive integer
- So the Time Complexity for running the first loop **is O(n * Time Complexity of the second for loop)**.
- The rows.append([]) binding that is inside the first for loop, is appending a list to the rows list.
  - Because the appended list is empty, the time complexity for this operation is constant.
- The second for loop that is located within the first for loop will be executed **n** times.
- So the running Time Complexity of the second for loop is **O(n)**.
- The remainder rows[r].append(x) binding is constant.
- Therefore, the Time Complexity of the first for loop is:
  - O(n * Time Complexity of the second for loop)
  - O(n * O(n))
  - O(n^2)
- The overall Time Complexity is **O(n^2).**