CP2410 Practical 05 - Linked Lists

1. (R-7.1) Give an algorithm for finding the second-to-last node in a singly linked list in which the last node is indicated by a next reference of None.

2. (R-7.5) Implement a function that counts the number of nodes in a circularly linked list.

3. (R-7.6) Suppose that x and y are references to nodes of circularly linked lists,although not necessarily the same list. Describe a fast algorithm for telling if x and y belong to the same list.

4. Using PositionalList (ch07/positional_list.py) write a function **list_to_positional_list(L)** which takes a built-in Python list L and creates a new PositionalList containing the same elements in the same order.

5. (R-7.11) Implement a function, L.max(), that returns the maximum element from a PositionalList (ch07/positional_list.py) instance L. Assume all values in L are numbers.