Assumptions:
- List contains only whole positive integer values (i.e. i > 0)
- List is finite and unsorted
- List may contain duplicates
- List is not empty, at least 2 elements exist
- First and second element will be used for Max1 and Max2, respectively
    - Max1 is the largest value
    - Max2 is the second largest value
    - If Max1 = Max2 when initializing
        - This problem should fix itself if any element in the list satisfies the
          Max1 or Max2 condition(s)
        - If still same when outputting results, same Max1 and Max2 values implies
          there was only one maximum value in the List
- When looping:
    - Max1 condition is check first
    - Max2 condition(s) is/are check second
    - else: Max1 and Max2 not updated
- Must loop through List only one time
- Left to the reader to implement code

Find Top Two Max Values

Max1 = A.get(0)

Max2 = A.get(1)

[A.hasNext()]

[Max1 < A.get(i)]

[Max2 < A.get(i) < Max1]

move old Max1 into Max2

update Max2

update Max1 with A.get(i)

[else]

[else]

i++

Output Max1 and Max2