WangQiuc ★ 1552 Last Edit: June 26, 2019 8:36 PM 679 VIEWS

15

Suppose we have such a linked list, with L-long line concatenating with a C-long cycle. And initially we have two pointers **a** and **b** pointing at the head of the linked list. **a**'s stride is 2 and **b**'s stride is 1. If there is a cycle, **a** will meet **b** again in the cycle since **a** will start to catch up with **b** when **b** enters the cycle (later than a) one step each time (a'stride-b'stride=1).

And suppose **a** and **b** meet at point X and the entry of the cycle is E. By saying |EX|=D (in forwarding direction), **b** has moved a distance of L+D while **a** has moved a distance of L+D+KC where K (K>0) is the times that **a** has been cycling.
Since **a**'s stride is the double of **b**'s, we have **L+D+KC = 2L+2D** or **L+D=KC**. So **L = C-D +(K-1)C**.

Now |XE| is what left for **b** to reach E (cycle's entry) again. Remember |EX|=D, so |XE|=C-D. Thus, if **b** moves a distance of L, which is C-D +(K-1)C, it will be at E. And if we have another pointer **c** move simultaneously with **b** but start at the head of linked list, **c** will walked through the line whose length is L and also reach the entry point E. So **b** and **c** will meet there, or their meeting point is the entry of the cycle.

D