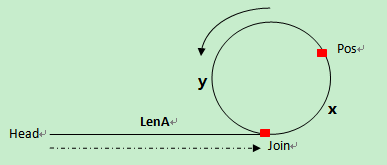
1. 怎么寻找单链表中环（环的起点）



定义两个指针p、q，p每次后移两个结点，称为快指针，q每次后移一个结点称为慢指针。假设pos为p、q相遇的结点，则此时q还未走完整个链表。

1. 证明快、慢指针相遇后，由相遇点开始慢指针向后移动LenA个结点到达环起点

慢指针移动了（LenA+x）个结点，则快指针移动了2\*（LenA+x）个结点，假设快指针延环走了n圈，则有n>=1。则可列：

**LenA+x = 2\*(LenA+x) – n\*(x+y)，**

化简有：

**n\*(x+y) = LenA+x，**

等式两侧同时加上join起点有：

**Join + n\*(x+y) = join +x+LenA**

又因为x+y为环一周的长度，所以等式左侧=join。join+x即为快慢指针相遇点Pos, 则有：

**Join = Pos+LenA，**

得证，由相遇点开始，慢指针向后移动LenA个结点到达环的起始结点。

1. LenA未知，如何得到Join结点？

注意，由上述证明可知Pos后移LenA个结点可到达Join，则由LenA==y。算法主要实现：

当快慢指针第一次相遇时，即都位于Pos位置，则使p指向链表头部，再将p、q逐步后移，则p、q会再Join处再次相遇，则得到环起点。

1. 并查集

3、字符串压缩