```
(1) Big problem divided into Similar Subproblems.
   (2) Focus on solving the subproblems instead of the whole scope.
2. Hon Recursion?
   (1) Wharis the Subproblem? >> Do what on Current level?
      Tail Recursion

Non-Tail Recursion

Convent level

Pars Parameter

Never level

Never level

Recursion

Never level

Recursion

Recursion

Never level

Recursion

Recursion
   (2) What represents next subproblem? >> Pass what into next level?
       Tait Recursion's main idea: Skip repeated operation, just can the method reself, pass new prameters at each con.
        Non-Tail Reemsion's main idea: Assuming all following levels are already handled, with verson value.
3. 3-Step:
    slep 1: base case l'exit, or return)
   slep 2: current layer ( core logic)
    seep 3: next layer
 Examples of Recursion
 Reverse LinkedList_Tail Recursion
             head
                         head.next
            Lise Node prev = num;
            Lise Node nexe = mm;
            white (head!= mm) }
                                      head. next
                           head
                                      Inexall
                                                                               O 结 head. next 极精身
                                                                                    next = head, next;
            head. next
                           head
                                                                               ②断升原来 To head > head. Nexte in 指向
                                                                                  à head, next this prev.
           head, next
                                                                                   head. next = prev;
                           head
                                       nexe
                                                                                   keep prer moving forward to do traverson
                                                                                   of the linkedlist, prer's next node is head"
                                                                                     prer = head;
                                        nexe
                head. next
                    null
                                                                                   keep head moving forward to do traverson
                                                                                   of the Linkedling, head's next mode is next"
                  head. West prev
                                                                                     head = nexte;
                                        nexe
                                                                                   这中写"和和和和"
                 recum prev;
                                                                                 Tail Reunsian
                      Veración
                    if ( head == mm | | head. next == mm) {
                                                                                                 head. next
                                                                                      head
                                                                              prev
                          recum head;
                                                                              null
                     Lise Node prev = num;
                                                                              privale Node reverse Helper (Node head, Node prev) {
                     List Node next = mm;
                     White (head != mm) }
                                                                                   if ( head == mm) {
                          // Reverse
                                                                                     vecum head;
                           next = head, next;
                            head, hext = prev;
                                                                                   Node next = head. next.
                         11 Prepare for next iteration
                                                                                   head. next = prev.
                           prer = head;
                                                                                   rerum reverseHelper (next, head);
                            head = next;
                       ream prev;
Reverse LinkedList_Non-Tail Recursion
                        head. next
            head
                                                                                                public Node reverse Reunsière (LiseNode head) }
                                                                                                       11 Base Case
           编篇 head ithin 部为都已经 reverse 知了.
                                                                                                         if (head = mull 11 head. next == mul) {
                                                                                                              voern head;
                          head. next
           head
                                head, next, next
                                                                                                         List Node neu Head = reverse Recursive (head, vext);
            le of, head 指向"1", A head. Next 指向 2, 他 head. Next. Next 指向 null
                                                                                                          11 Reverse
             雷伽如事:
                                                                                                           head. next, next = head;
                         1 Let head, herr, next this head
                                                                                                           head. next = mull;
                                                                  hewHead
                                              head. next
                              head
                                                                                                           remin new Head;
                                                   Itthead, next, next
                         D'Ler head. Next Fhis nuu
                                                                hewHead
                            head
                                    (2)
```

hewHead

head

head. hext

nu

3 return new Head

SlidesPractice_L2_Recursion_Diagram

Notes for Recursion

1. Why Recursion?