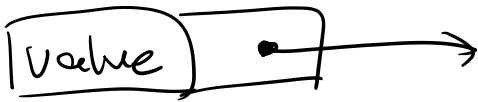
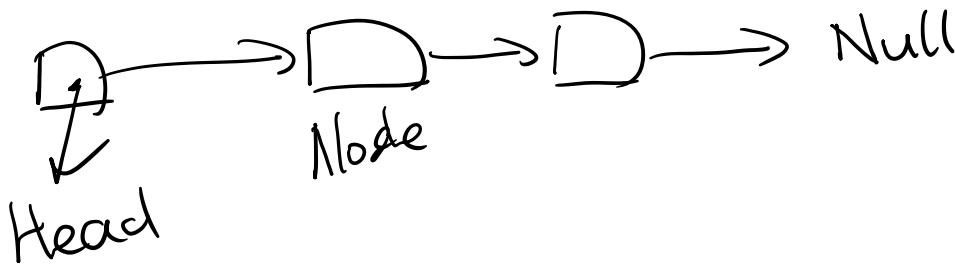


Data structure

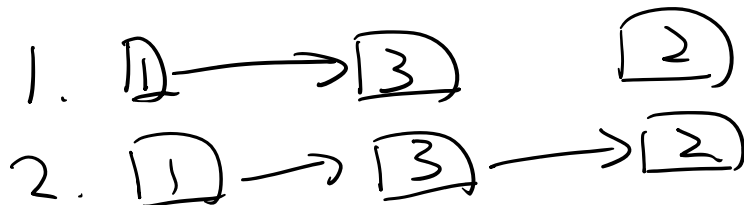
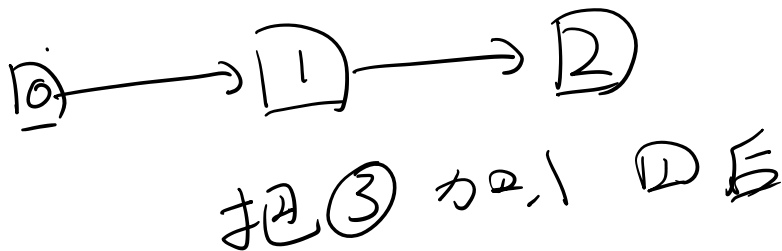
December 5, 2016 2:13 PM

Single 1. Linked List.

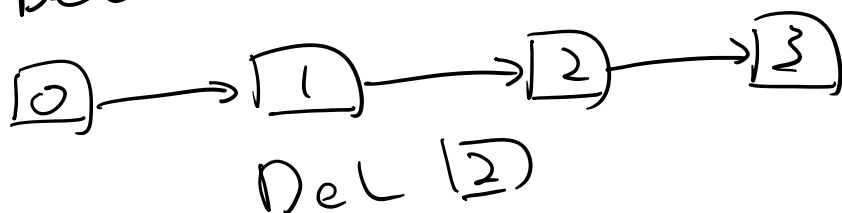
Node  pointer to next.



a) Insert.

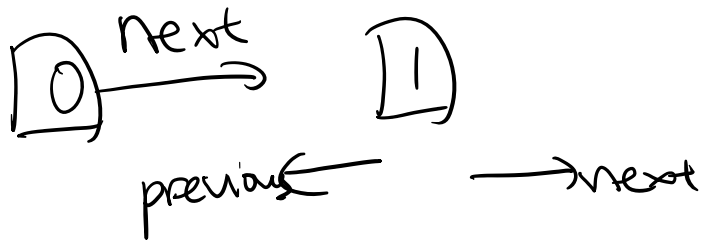


b) Del

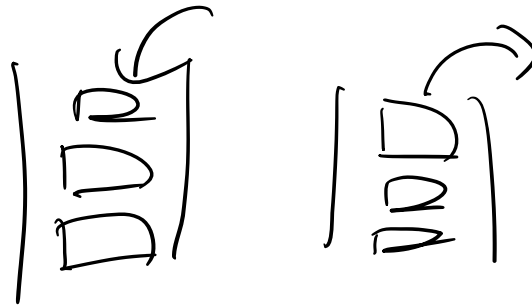


$1.next = 1.next.next$

2. Double.



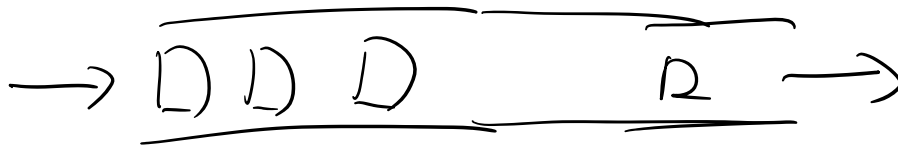
2. Stack. Last In First Out



push () :

pop () :

3. Queue: First In First Out



enqueue () :

dequeue () :

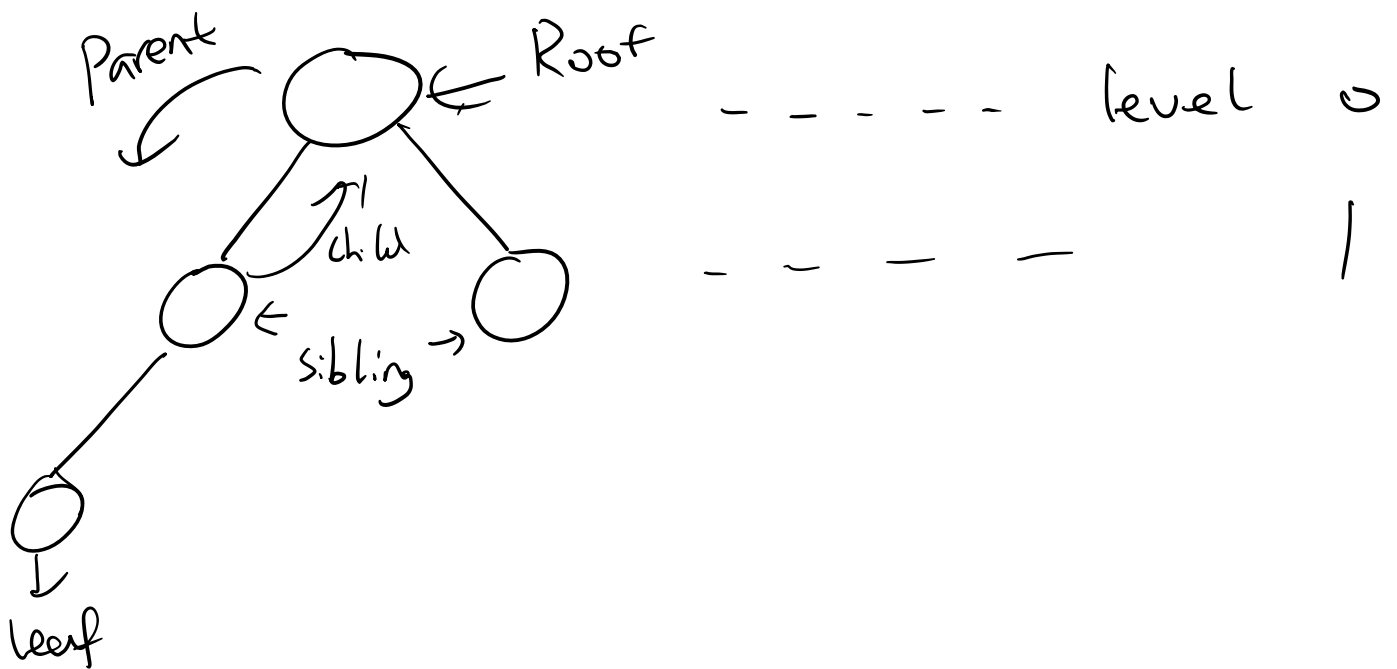
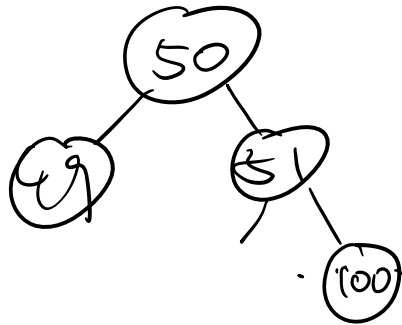
push () :

pop () :

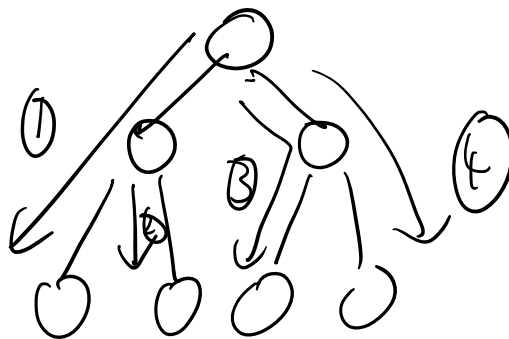
4. BST.

Search an item in a list

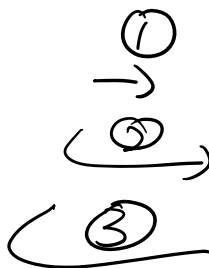
1 2 ... 100



DFS.



BFS



递归 1) 递归.

Factorial:

$$3! = 3 \times 2 \times 1$$

$$10! = 10 \times \dots \times 1.$$

Def fac(n):

if $n == 1$: return 1 \Rightarrow 首先 Base.
 else : return $n \times \text{fac}(n-1)$.

$n=3$:

① $3 \times \text{fac}(2)$

② $2 \times \text{fac}(1)$

③ $3 \times 2 \times 1 = 6$. 不一定好. 简单.

Fibonacci:

0, 1, 1, 2, 3, 5, 8, ...

$$F_0 = 0, F_1 = 1, F_2 = 1, F_3 = 2, \dots$$

Def Fib(n):

if $n == 0$, return 0.

else if $n == 1$: return 1:

else: return $\text{Fib}(n-1) + \text{fib}(n-2)$

$n = 4$.

① $\text{Fib}(3) + \text{fib}(2)$

② $\text{Fib}(1) + \text{Fib}(2) + \text{Fib}(2)$

$$1 + \overset{\text{Fib}(1+1)}{1} + 1 = 3$$

