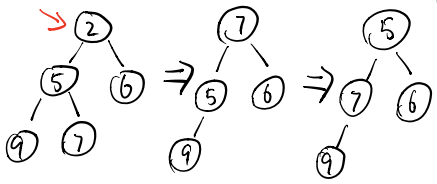
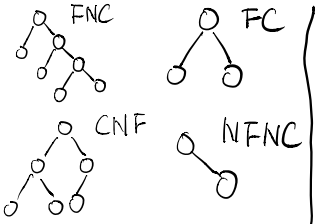
delMin in Heap: last node



Binary tree is:

full: if each node have two child nodes.

complete: 1. completely full
2. last node on left



Graph:

Dijkstra's Algorithm: start node 为 0，其他为 ∞，找最小，过去影响现在

Prim's Alg: 从 start node，找最小 cost 边

Kruskal's Alg: 先排序，从小→大加入

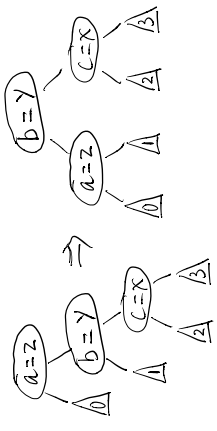
Boruvka's Alg: 把 node 当树，用最小边连接两个树.

注意：graph 不能出现 cycle.

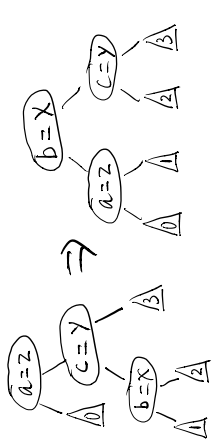
AVL-tree:

x's grandparent z is unbalanced node

1.

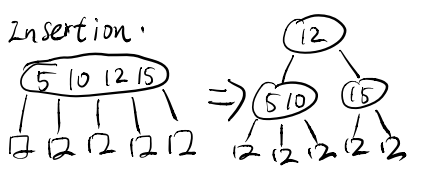


2.

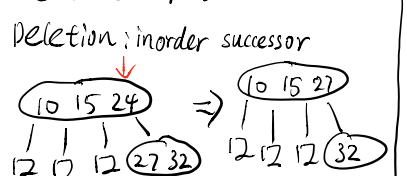


(2,4) tree: all external nodes have same depth

Insertion:

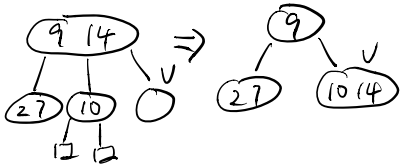


Deletion: inorder successor

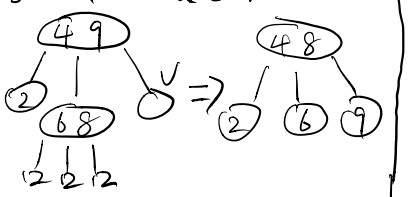


underflow:

1. adjacent siblings of v are 2-children.



2. adjacent siblings of v are 3 or 4-children.



Flow Network:

Flow $f(X)$ across a cut X: total flow of forward edge - total flow of backward edge.

Capacity $c(X)$ of cut X: total capacity of forward edges.

min-cut:

1. forward edge: $f(e) = c(e)$
2. Backward edge: $f(e) = 0$

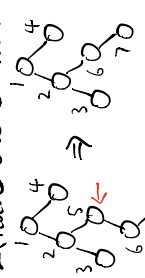
Ford-Fulkerson Alg:

Bipartite matching

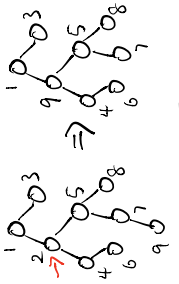
Deletion in BST:

1. leaf: 直接删

2. have one child:



3. otherwise: inorder successor



Dynamic Programming

$$B[k,w] = \begin{cases} B[k-1,w] & \text{if } w < w_k \\ \max\{B[k-1,w],\ b_k + B[k-1, w-w_k]\} \end{cases}$$

i	1	2	3	4
bi	25	15	20	36
wi	7	2	3	6

	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	25	25	25	25
2	0	0	15	15	15	15	15	25	25	40	40
3	0	0	15	20	20	35	35	35	35	40	45
4	0	0	15	20	20	35	36	36	51	56	56

root 1: child of A[i]: A[2i], A[2i+1]

root 0: child of A[i]: A[2i+1], A[2i+2]

parent of A[i]: A[⌊(i-1)/2⌋]

等比: $S_n = \frac{a_1(1-q^n)}{1-q}\ (q \neq 1)$

等差: $S_n = \frac{n(a_1+a_n)}{2}$

$\sum_{k=0}^{n-1} ar^k = a\left(\frac{1-r^n}{1-r}\right)$

$\sum_{k=1}^{n} k^2 = 1^2 + 2^2 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$

$\sum_{k=1}^{n} k(k+1) = 1\times2 + 2\times3 + \cdots + n\times(n+1) = \frac{n(n+1)(n+2)}{3}$

n 个节点的二叉树有 $f(n) = \frac{(2n)!}{n!\,(n+1)!}$ 个状态.

$\log_a(MN) = \log_a M + \log_a N$

$\log_a b = \frac{\log_c b}{\log_c a}$

$\log_a b \cdot \log_b a = 1$

$\log_a(1/N) = -\log_a N$

$\log_a M^n = n\log_a M$

$mid = \lfloor (low + high)/2 \rfloor$

The master method

$$T(n) = \begin{cases} c & \text{if } n < d \quad d \geq 1,\ a > 0, \\ aT(n/b) + f(n) & c > 0,\ b > 1 \end{cases}$$

1. If $f(n) = O(n^{\log_b a - \varepsilon}), \varepsilon > 0$, $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta(n^{\log_b a})$, $T(n) = \Theta(n^{\log_b a} \lg n)$
3. If $f(n) = \Omega(n^{\log_b a + \varepsilon}), \varepsilon > 0$, if $af(n/b) \leq cf(n), c < 1$, $T(n) = \Theta(f(n))$

1. applies where $f(n)$ is polynomially smaller than special function $n^{\log_b a}$
2. asymptotically close, 3. poly-larger

$f(n)$ is asymptotically smaller than $g(n)$ if $f(n) = O(g(n)/n^{\varepsilon})$ for $\varepsilon > 0$.

$f(n)$ is ... ... larger that $g(n)$ if $f(n) = \Omega(g(n)n^{\varepsilon})$ for some $\varepsilon > 0$.

Euclidean Algorithm

Fact: $\gcd(a,0)=a$

Let a, b, q, and r be integer such that $a=bq+r$, and $b\neq 0$, then $\gcd(a,b)=\gcd(b,r)$

Basic Euclidean Algorithms:

```
def gcd(a,b)
assert a≥b and b≥0 and a+b>0
return gcd(b, a%b) if b>0 else a
```

Extended Euclidean Alg: $s\times a+t\times b=\gcd(a,b)$

```
r1←a, r2←b, s1←1, s2←0, t1←0, t2←1
while (r2>0) { q←⌊r1/r2⌋
r←r1−q×r2; r1←r2; r2←r;
s←s1−q×s2; s1←s2; s2←s;
t←t1−q×t2; t1←t2; t2←t; }
gcd(a,b)←r1; s←s1; t←t1
```

---

b^{-1}:

```
r1←n; r2←b; t1←0; t2←1;
while (r2>0) { q←⌊r1/r2⌋;
r←r1−q×r2; r1←r2; r2←r;
t←t1−q×t2; t1←t2; t2←t; }
if (r1=1) then b^-1 = t1
```

整除: $n\mid a$, 不可整除: $n\nmid a$, a/n

1. if $a\mid 1$, then $a=\pm 1$
2. if $b\mid a$ and $a\mid b$, then $a=\pm b$
3. if $b\mid a$ and $c\mid b$, then $c\mid a$
4. if $a\mid b$ and $a\mid c$, then $a\mid(m\times b+n\times c)$

---

1. $(a+b)\bmod n=[(a \bmod n)+(b \bmod n)]\bmod n$
2. $+, -, \times$
3. $10^n \bmod x=(10 \bmod x)^n \bmod x$

Inverse:

Additive inverse: $a+b\equiv 0 \pmod n$

multiplicative inverse: $a\times b\equiv 1 \pmod n$

no multiplicative inverse if $\gcd(a,b)=2\neq 1$

$\gcd(a,b)=1 \Rightarrow a$ and b are relatively prime

Z_n: $Z_6=\{0,1,2,3,4,5\}$, Z_n^*: Z_n 中所有与 n 互质的数

Let p be prime, and let x be an integer that $x \bmod p\neq 0$, then $x^{p-1}\equiv 1 \bmod p$

$x^{-1}\equiv x^{p-2} \pmod p$

Euler's function: $\varphi(n)$ 是 Z_n^* 的长度:

$n=p_1\times p_2\times\cdots\times p_n$, $\varphi(n)=n\times(1/p_1)\times\cdots\times(1/p_n)$

p 为质数, 且相同的 p 只算一次。

Let n be a positive integer, and x be integer that $\gcd(x,n)=1$, then $x^{\varphi(n)}\equiv 1 \pmod n$

RSA: plaintext: 明文 ciphertext: 密文

Let $n=p\cdot q$ and $\varphi(n)=(p-1)(q-1)$

e and $\varphi(n)$ are relatively prime,

$ed\equiv 1 \pmod{\varphi(n)}$; p, q are prime.

public key: n, e, private key: d

Encryption: $C=M^e \bmod n$

Decryption: $M=C^d \bmod n$

Digital signatures:

Senter: $S\leftarrow M^d \bmod n$

recer: $M\equiv S^e \pmod n$.