一、

(1)、初始化：在循环开始之前它为真

保持：循环在一次迭代前为真，在下一次迭代之前仍为真

终止：循环终止的时候，提供一个可以用来证明算法正确性的性质

(2) 动态规划：通过求子问题的解来求解原问题（子问题会重叠），并且把子问题的解存在表中，这样只用求一次

实现方法：自底向上法和带备忘的自顶向下法

(3) 平衡二叉树：每个节点最多有两个子节点，每个节点的值比左子树的所有节点的值大，比其右子树所有的节点小，左子树高度与右子树高度差小于等于1

红黑树五个特点：1、节点颜色只为黑色或红色

2、根节点是黑色

3、红节点的子结点和父节点是黑色

4、叶子节点为黑

5、一个节点到叶子节点的每个路径中，黑节点个数同等

二、

4、 $4n^2 \le 128n\lg n$　　$n \geq 2$　　A优于B

5、 ① $a=2$　$b=2$　$f(n)=n^4=\Omega(n^{\varepsilon+1})$　$\varepsilon=3$

$n^{\log_2 2}=n$　$\frac{f(n)}{n}=n^3$

$2f(\frac{n}{2})=\frac{n^4}{8}\le cn^4 \xleftarrow{成立} \frac{1}{8}\le c<1$

$\therefore T(n)=\Theta(n^4)$

② $a=16$　$b=4$　$n^{\log_4 16}=n^2$　$\frac{n^2}{n^2}=1$

$T(n)=\Theta(n^2\lg n)$

③ $T(n) = T(n-2) + n^2$

$T(n-2) = T(n-4) + (n-2)^2$

$T(n) = T(n-4) + (n-2) + n^2 = T(n-6) + (n-4)^2 + (n-2)^2 + n^2$

$= T(n-8) + (n-6)^2 + (n-4)^2 + (n-2)^2 + n^2$

$\therefore = n^2 + (n-2)^2 + \cdots \quad T(2)$

$= \underset{\downarrow \atop C}{T(2)} + \sum_{i=0}^{\frac{n}{2}-2} (n-2i)^2 \sim n^3$

$= \Theta(n^3)$

④ $a=7 \quad b=2 \quad f(n)=n^2 \quad n^{\log_b a} = n^{\log_2 7}$

$\Theta(n^2) = O(n^{\lg 7 - \varepsilon}) \quad \varepsilon = 0.807$

$T(n) = \Theta(n^{\lg 7})$

⑤ $T(n) = T(n-1) + n = T(n-2) + n - 1 + n = T(n-3) + (n-2) + (n-1) + n \cdots$

$= T(0) + 1 + 2 + 3 + \cdots (n-2) + (n-1) + n$

$= 1 + 1 + 2 + 3 + \cdots n$

$= 1 + \frac{(1+n) \times n}{2} = O(n^2)$

6、 $f(n) \in O(g(n)) \quad \therefore \exists$ c和$n_0 > 0$ s.t 当 $n > n_0$ 时

$0 \leq f(n) \leq c\,g(n)$

假设 $f(n) \in \omega(g(n))$

则 $\exists\ n_0', c' > 0$ s.t $n > n_0'$ 时 $0 \leq c' g(n) < f(n)$

此时不论 n_0' 和 n_0 的大小，在n趋于无穷时无法存在 c', c, n_0, n_0' 使

$f(n) \leq c\,g(n)$

$c' g(n) < f(n)$ 同时成立

∴ 假设错误

∴ $f(n) \notin \omega(g(n))$

三、

7、 $P(X=1)=p \quad P(X=0)=1-p \quad alg$

在alg的基础上 运行两次，得 0,1 / 1,0 > 概率同 都为 $p(1-p)$

1,1 / 0,0 舍

结果 (0,1) 为模拟1，(1,0) 模拟0，(1,0)，(0,1) 发生是等可能的

期望时间：

8、 仿 Strassen

$$S_1=(a+b)\times c=ac+bc$$
$$S_2=(c+d)\times b=bc+bd$$
$$S_3=(b-a)\times d=bd-ad$$
$$\therefore\ ac-bd=S_1-S_2$$
$$ad+bc=S_2-S_3$$

9、

(1) $f(n)\le \max(f(n),g(n))$

$g(n)\le \max(f(n),g(n))$

$\therefore\ f(n)+g(n)\geq \max(f(n),g(n))$

$\frac{f(n)+g(n)}{2}\le \max(f(n),g(n))$

$\therefore\ \max(f(n),g(n))\le f(n)+g(n)\le 2\max(f(n),g(n))$

$\therefore\ f(n)+g(n)=\Theta(\max(f(n),g(n)))$

(2) if $f(n)=O(g(n))$

则 $\exists$ c 和 n_0 s.t $n\ge n_0$ 时 $f(n)\le c\,g(n)$

$\therefore\ g(n)\ge \frac{1}{c}f(n)$

即 $\exists$ $c'=\frac{1}{c}$，n_0 s.t $n\ge n_0$ 时

$0\le c'f(n)\le g(n)$

$\therefore\ \Omega(f(n))=g(n)$

四

10、① 将S升序排列

② 给两个数组 a[] 和 b[]

a[0] b[n-1]

循环算 a[i]+b[k] 的和

i初=0 k初=n-1

若 a[i]+b[k]<X

i++

若 a[i]+b[k]>X

k--

若 a[i]+b[k]==X

reture true

若 i>k

reture false

11、初始化：$i=p-1, j=p$

保持：① $a[j]>x$ 时 $j{+}{+}$，$\underline{a[j-1]>x}$ ∴ 对 $i+1\le k\le j-1$ 都有 $a[k]>x$ ✓

② $a[j]\le x$ $i{+}{+}$，$a[p\cdots i-1]\le x$ + $a[i]\le x$ ∴ $a[p\cdots i]\le x$ 成立

终止：循环停止时，$j=r$ $a[p..i]\le x$ $a[i+1\cdots r-1]>X$

12、

a i | p j 13 19 3 6 4 8 11 18 | r 9

b i | p 13 | j 19 3 6 4 8 11 18 | r 9

c i | p 13 19 | j 3 6 4 8 11 18 | r 9

d p i 3 | 19 13 | j 6 4 8 11 18 | r 9

e p 3 i 6 | 13 19 | j 4 8 11 18 | r 9

f p 3 6 i 4 | 19 13 | j 8 11 18 | r 9

g p 3 6 4 i 8 | 13 19 | j 11 18 | r 9

H p 3 6 4 i 8 | 13 19 11 | j 18 r 9

i 3 6 4 i 8 | 13 19 11 18 | j r 9

j $\underbrace{3\ 6\ 4\ 8}_{<9}$ | 9 | $\underbrace{19\ 11\ 13}_{>9}$

13、

每组3个元素，至少有 $\frac{2n}{3}+4$ 个元素进递归

$T(n) \le T(\frac{n}{3}) + T(\frac{2n}{3}) + O(n)$

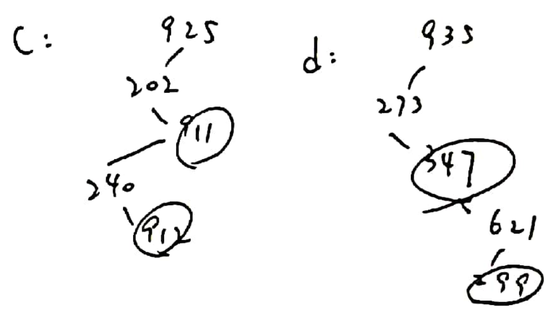
无法在线性时间内完成

五

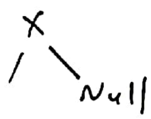
16、

(c)中 912 > 911

(d)中 299 比 (347) ∴ 不可能



17、



Y一定是X的祖先，不是X的后继

Y的左孩子也是X的祖先，如果不是则 x>y x

假设y不是x的最底层祖先，则用z表示x的最底层祖先，z在y的左子树中 z<y 与"x有一个后继y"矛盾

∴ Y是X的最底层祖先，Y.left也是X的祖先

18、
A_1 5×15
A_2 15×12
A_3 12×5
A_4 5×5

$A_1 (A_2A_3) A_4$

19、 每(i,j)对应一个点，$1 \le i \le j \le n$ →对应子问题

有 $\frac{n(n-1)}{2}+n$ 个点　　$\sum_{k=0}^{n-1} 2k(n-k)$ 条边

~~(i,j) → (k,l)~~

~~如果 k=i 且 k≤l<j~~

~~l=j 且 i<k≤j~~

20、 $q = \max(q, p[i] + r[j-i] - 2)$

↓

减去切割开销2