1. C语言允许连续赋值。因此F题可以a[x1][y1] = a[x2][y2] = ‘/’。
2. 2的多少次方的乘除，尽量用位运算而不是pow函数，省时且简洁。（例： 1 <<(n-1) 表示把二进制下的1左移n-1位， 也就是pow(2,n-1)。）
3. 有浮点数一定注意。一是不要不小心把浮点数写成int； 二是浮点数的计算天然有误差，如果需要尽可能精确的话需要注意算法（IEEE 754标准）；三是浮点数的绝对值用fabs()函数（<math.h>）（不能和0比较然后负的话取反，原因为第二条）。
4. 递归一般比较耗时间，链式的线性递归还好说，如果一个递归函数里面>=2次（比如x次）调用自己来递归，那计算量会指数级暴增(约pow(x,n)级别)。所以递归一定尽可能用比较简单的算法来实现，而且如果需要多次调用这个递归函数一定要开一个变量存储它的结果，这样后续只需要用这个变量的值就可以，一定不要出现多少次调用多少次。（其它也是，比如多次需要pow(x,y), 就令a = pow(x,y), 后续再用就用a。）