六道腾讯、百度、美团常爱问的面试智力题和答案

赛马次数

有 25 匹马和 5 条赛道，赛马过程无法进行计时，只能知道相对快慢。问最少需要几场赛马可以知道前 3 名。

先把 25 匹马分成 5 组，进行 5 场赛马，得到每组的排名。再将每组的第 1 名选出，进行 1 场赛马，按照这场的排名将 5 组先后标为 A、B、C、D、E。可以知道，A 组的第 1 名就是所有 25 匹马的第 1 名。而第 2、3 名只可能在 A 组的 2、3 名，B 组的第 1、2 名，和 C 组的第 1 名，总共 5 匹马，让这 5 匹马再进行 1 场赛马，前两名就是第 2、3 名。所以总共是 5+1+1=7 场赛马。

A 组：1，2，3，4，5  
B 组：1，2，3，4，5  
C 组：1，2，3，4，5  
D 组：1，2，3，4，5  
E 组：1，2，3，4，5

用绳子计时 15 分钟

给定两条绳子，每条绳子烧完正好一个小时，并且绳子是不均匀的。问要怎么准确测量 15 分钟。

● 点燃第一条绳子 R1 两头的同时，点燃第二条绳子 R2 的一头；

● 当 R1 烧完，正好过去 30 分钟，而 R2 还可以再烧 30 分钟；

● 点燃 R2 的另一头，15 分钟后，R2 将全部烧完。

九球称重

有 9 个球，其中 8 个球质量相同，有 1 个球比较重。要求用 2 次天平，找出比较重的那个球。

将这些球均分成 3 个一组共 3 组，选出 2 组称重，如果 1 组比较重，那么重球在比较重的那 1 组；如果 1 组重量相等，那么重球在另外 1 组。

对比较重的那 1 组的 3 个球再分成 3 组，重复上面的步骤。

药丸称重

有 20 瓶药丸，其中 19 瓶药丸质量相同为 1 克，剩下一瓶药丸质量为 1.1 克。瓶子中有无数个药丸。要求用一次天平找出药丸质量 1.1 克的药瓶。

可以从药丸的数量上来制造差异：从第 i 瓶药丸中取出 i 个药丸，然后一起称重。可以知道，如果第 i 瓶药丸重 1.1 克/粒，那么称重结果就会比正常情况下重 0.1 \* i 克。

得到 4 升的水

有两个杯子，容量分别为 5 升和 3 升，水的供应不断。问怎么用这两个杯子得到 4 升的水。

可以理解为用若干个 5 和 3 做减法得到 4。

● 不能从 3 做减法得到 4，那么只能从 5 做减法得到 4，即最后一个运算应该为 5 - 1 = 4，此时问题转换为得到 1 升的水；

● 1 升的水可以由 3 做减法得到，3 - 2 = 1，此时问题转换为得到 2 升的水；

● 5 - 3 = 2。

5升水盛满，倒入三升水的杯子，那么还剩2升水。

将3升水的杯子水全部倒掉，然后把5升水的杯子里面还剩的2升水倒入3升水的杯子，杯子差一升水不满。

将5升水倒满，然后倒入3升水杯子里面，直到倒满。能容纳1升水的大小，此时5升水的杯子里面就只剩4升水。

**高楼扔鸡蛋问题**

N个鸡蛋从M楼层摔（2个鸡蛋从100层摔）

一、题目：

　　有一栋楼共100层，一个鸡蛋从第N层及以上的楼层落下来会摔破， 在第N层以下的楼层落下不会摔破。给你2个鸡蛋，设计方案找出N，并且保证在最坏情况下， 最小化鸡蛋下落的次数。

二、思路：

先假设，最小的次数为x次。

首先在x层摔，那么会出现两个结果：

1、碎了，为了找出那一层碎了，第二个鸡蛋必须从1~x-1进行遍历的摔

2、没碎，那么第二次就在x+(x-1)楼层摔。

为什么是x+x-1楼层呢？

首先我们已经假设了通过x步我们就能得到答案，现在我们在x层已经用了一次了，那么就只剩下x-1步了。所以我们选择x+(x-1)层，如果碎了，我们就能通过x-2步，遍历x+1~x+(x-1)-1的所有楼层。

3、如果在x+(x-1)楼碎了，那么同1，遍历x+1~x+(x-1)-1

4、没碎，那么同2，就在x+(x-1)+(x-2)层摔

…

最后我们将会得出这样一个楼层公式x+(x-1)+(x-2)+…+1 = x(x+1)/2。

这个公式有什么意义呢？

有， x(x+1)/2 >= 100，这样才能顺利的解除x。

有人说，x(x+1)/2 = 99就可以，如果鸡蛋在99层都没碎，那么必定是100层。 我想说谁告诉你一定会碎！

那么我们就顺利的解出 x=14。

三、扩展

此题还有一个扩展，就是为N个鸡蛋从M层摔找出最小值。

定义dp[i][j] 表示i个鸡蛋j层楼 找出的临界楼层值。

动态规划式子如下：

f[n][m] = 1+max(f[n-1][k-1],f[n][m-k]) k属于[1,m-1]

在第k层楼扔第一个鸡蛋。

碎了: 那么再扔的次数为 n-1个蛋，k-1层楼。一定在0~k-1中间。

没碎: 那么还是n个蛋，楼层收窄成m-k 需要继续测试k+1~m层楼。

解释下原理：

1、当手里有n个的时候，鸡蛋从k层往下摔，如果破了，那么手里只有n-1鸡蛋了，那么就需要测试f[n-1][k-1]楼层。或者更通俗好理解点的，我们运用2个鸡蛋100楼层的题目举例子。以上式子变为：f[2][m] = 1+max(f[1][k-1],f[2][m-k])

　　那么当手里有2个鸡蛋的时候，在k层摔，碎了。那么现在手里也就只有一个鸡蛋了，此时我们必须遍历1~k-1找出第一次碎的楼层。所以为1+f[1][m-k]，前面的1代表在k层的操作。

2、没破，那么手里还有n个鸡蛋，那么需要测试k+1~m这些楼层。

此时我想问下，当手里有2个鸡蛋测试1~m-k层和手里有2个鸡蛋测试k+1~m有什么区别？

有人说有，因为楼层越高越容易碎，那其实是你个人的想法罢了。其实并没有区别，所以第一个公式可以写为f[n][m-k]。

最后附上代码，为了理解方便，而不必从数组从0开始而困扰，这里就空间多开了点，所以如果拿去用的话，可以优化下：

public class Eggs{

public int countMinSetp(int egg,int num){

if(egg < 1 || num < 1) return 0;

int[][] f = new int[egg+1][num+1];//代表egg个鸡蛋，从num楼层冷下来所需的最小的次数

for(int i=1;i<=egg; i++){

for(int j=1; j<=num; j++)

f[i][j] = j;//初始化，最坏的步数

}

for(int n=2; n<=egg; n++){

for(int m=1; m<=num; m++){

for(int k=1; k<m; k++){

//这里的DP的递推公式为f[n][m] = 1+max(f[n-1][k-1],f[n][m-k]) k属于[1,m-1]

//从1-m层中随机抽出来一层k

//如果第一个鸡蛋在k层碎了，那么我们将测试1~k-1层，就可以找出来，也即1+f[1][k-1]

//如果第一个鸡蛋在k层没有碎，那么我们将测试k+1~m也即m-k层，

// 这里也是重点！！！！

// 现在我们手里有2个鸡蛋，要测试m-k层，那么我想问，此时和你手里有2个鸡蛋要测试1~m-k层有什么区别？

// 没有区别！是的在已知k层不碎的情况下，测试k+1~m层的方法和测试1~m-k没区别，所以可以写成 1+f[n][m-k] 其中1表示为 在k层的那一次测试

f[n][m] = Math.min(f[n][m],1+Math.max(f[n-1][k-1],f[n][m-k]));

}

}

}

return f[egg][num];

}

public static void main(String[] args) {

Eggs e = new Eggs();

System.out.println(e.countMinSetp(2,100));

}

}

参考文章：

100层楼2个鸡蛋，如何得知鸡蛋能承受几层的撞击

两个鸡蛋100层楼(DP)