从圆桌问题谈数据结构的综合运用

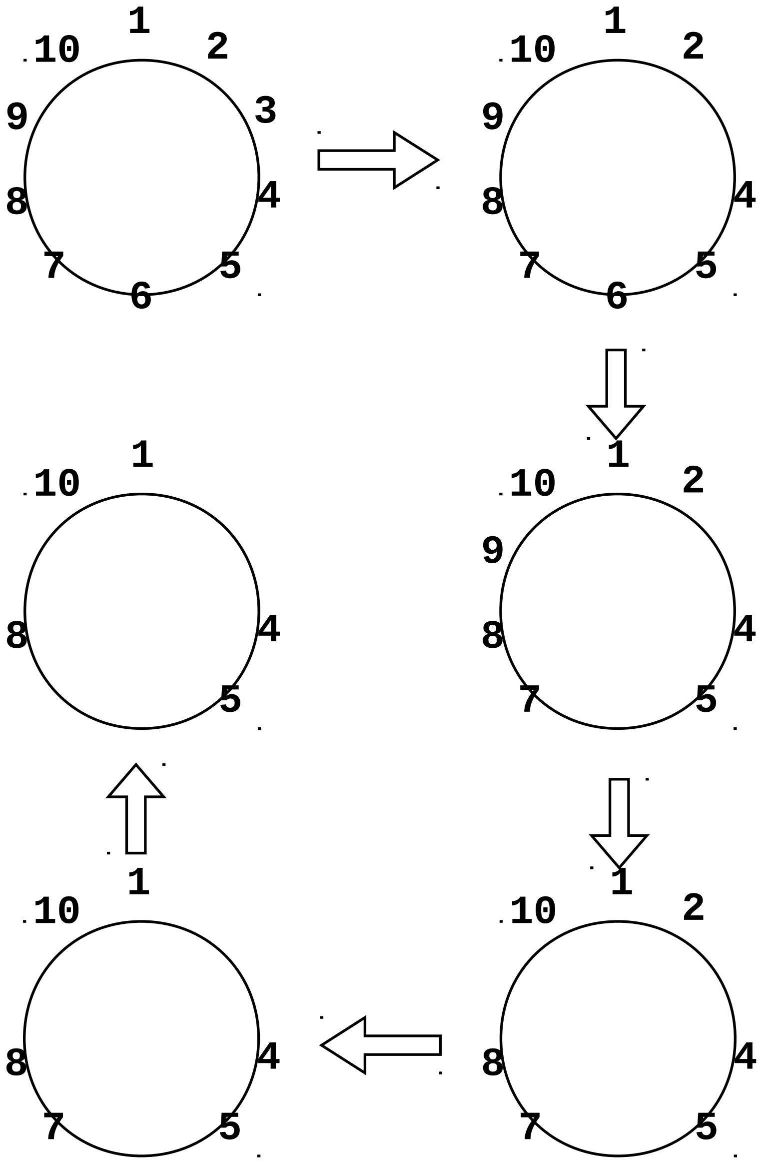
圆桌问题

题目：圆桌上围坐着 2n 个人。其中 n 个人是好人，另外 n 个人是坏人。如果从第一个人开始数数，数到第 m 个人，则立即处死该人；然后从被处死的人之后开始数数，再将数到的第 m 个人处死…依此方法不断处死围坐在圆桌上的人。试问预先应如何安排这些好人与坏人的座位，能使得在处死 n 个人之后，圆桌上围坐的剩余的 n 个人全是好人。

输入：文件中的每一行都有两个数，依次为 n 和 m，表示一个问题的描述信息， $n\leq32767$，$m\leq32767$。

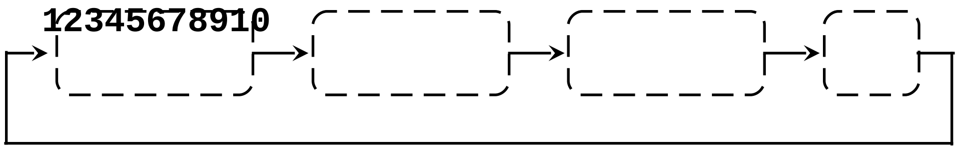
输出：依次输出每一个问题的解。每一个问题的解可以用连续的若干行字符来表示，每行的字符数量不超过 50。但是在一个问题的解中不允许出现空白字符和空行，相邻的两个问题的解之间用空行隔开。用大写字母 G 表示好人，大写字母 B 表示坏人。

圆桌问题实现思想图示（$n=5,m=3$）

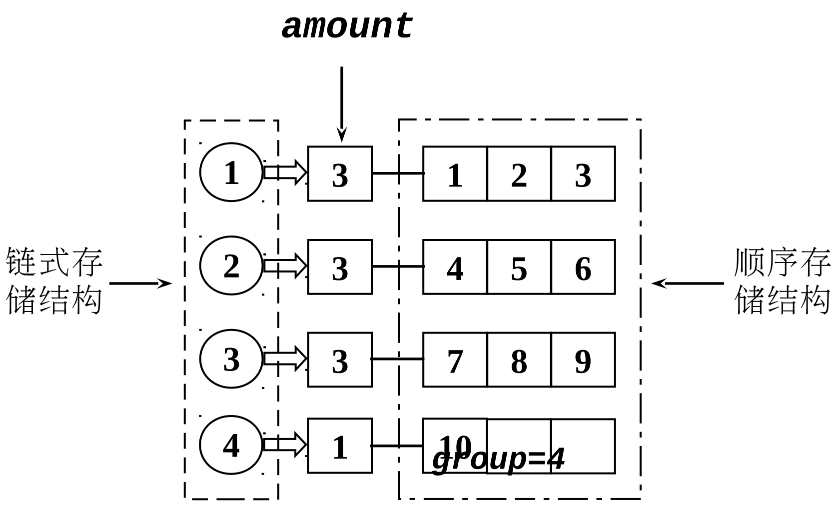


分段式数据结构示意

(思想模型)



(实际模型)



共进行 **1+2+2+3+5=13** 次操作

改进前后程序效率比较

（测试机器：*P166*）

测试数据	线性表“查找”法	“优化直接定位”法	
		amount=400	改进前用时是改进后的多少倍
n=200　m=100	0.000s	0.000s	／
n=1000　m=50	0.440s	0.000s	／
n=32767　m=200	5.870s	0.930s	6.312
n=32767　m=1000	29.440s	0.980s	30.041
n=32767　m=10000	294.120s	1.260s	233.43
n=32767　m=20000	588.530s	1.590s	370.14
n=32767　m=32767	963.560s	1.970s	489.12

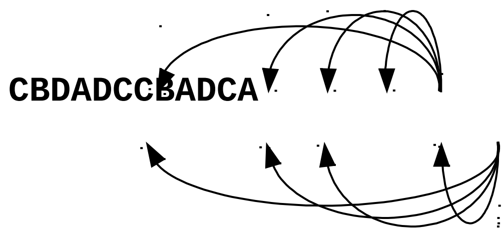
引申

➢ **横向延伸——约瑟夫环类的问题**

如：《翻牌游戏》、《猴子选大王》

➢ **纵向延伸——数据结构的综合运用**

在解决一些数据规模较大的问题时有很好的效用。如《隐藏的码字》（*IOI'99*）。在解决这道题目时，如果建立起链式和顺序相结合的数据结构（如下图），程序效率就比较高。



链式和顺序相结合的数据结构实现简单，效果显著，应用比较广泛。当然还有其它的结合，比如二叉堆和顺序结构的一一映射（单射），在解决某些问题时会有很好的效果。

顺序存储结构操作示意

1	2	3	4	5	6	7	8	9	10	7
		☆								

1	2	4	5	6	7	8	9	10		4
				☆						

1	2	4	5	7	8	9	10			1
						☆				

1	2	4	5	7	8	10				5
	☆									

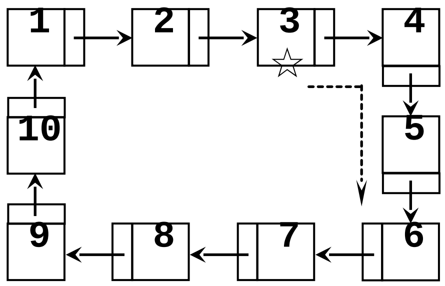
1	4	5	7	8	10					2
			☆							

1	4	5	8	10					

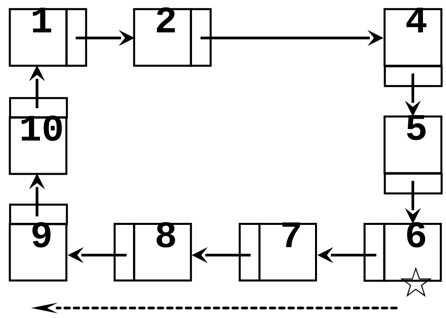
共进行 **7+4+1+5+2=19** 次操作，时间复杂度 $O(n^2)$。

链式存储结构操作示意

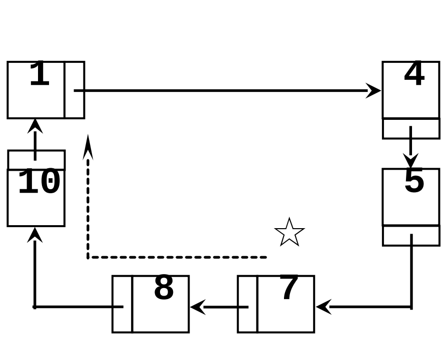
Step 1



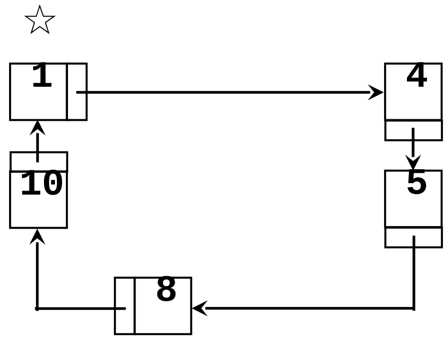
Step 2



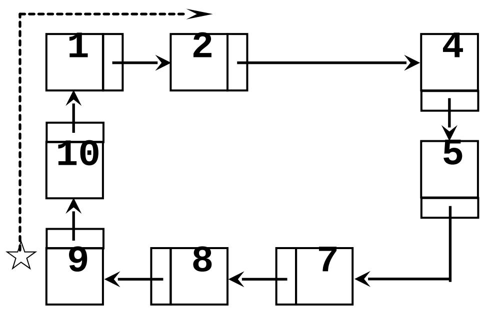
Step 3



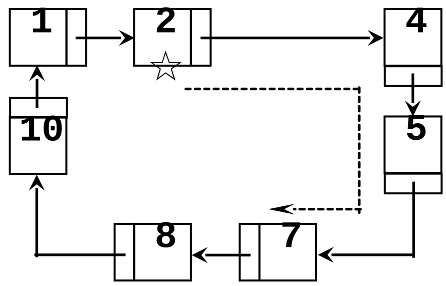
Step 4



Step 5



Step 6



共进行 **5×3=15** 次操作，时间复杂度 *O(nm)*。