**1666. 【AHOI2009】飞行棋 (Standard IO)**

Time Limits: **1000 ms**  Memory Limits: **65536 KB**  Detailed Limits

**Description**

　　给出圆周上的若干个点，已知点与点之间的弧长，其值均为正整数，并依圆周顺序排列。   
　　请找出这些点中有没有可以围成矩形的，并希望在最短时间内找出所有不重复矩形。

**Input**

　　第一行为正整数N，表示点的个数，接下来N行分别为这N个点所分割的各个圆弧长度

**Output**

　　所构成不重复矩形的个数

**Sample Input**

8

1

2

2

3

1

1

3

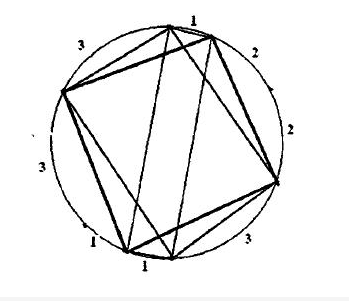
3

**Sample Output**

3

**Data Constraint**

**Hint**

　　N<= 20   
　　

**1110. CQOI2009循环赛 (Standard IO)**

Time Limits: **1000 ms**  Memory Limits: **65536 KB**  Detailed Limits

**Description**

n支队伍打比赛，每两支队伍恰好比赛一场。平局时各得1分，而有胜负时胜者3分，负者0分。  
假设三支队伍得分分别为3, 3, 3，则可能有两种情况：   
队伍 A B C 得分  
A - 3 0 3  
B 0 - 3 3  
C 3 0 - 3  
  
队伍 A B C 得分  
A - 0 3 3  
B 3 - 0 3  
C 0 3 - 3  
  
给出n支队伍的最终得分（即所有比赛均已结束），统计有多少种可能的分数表。

**Input**

第一行包含一个正整数n，队伍的个数。第二行包含n个非负整数，即每支队伍的得分。

**Output**

输出仅一行，即可能的分数表数目。保证至少存在一个可能的分数表。

**Sample Input**

样例输入1：

3

3 3 3

样例输入2：

2

0 3

样例输入3：

3

4 1 2

样例输入4：

6

5 6 7 7 8 8

**Sample Output**

样例输出1：

2

样例输出2：

1

样例输出3：

1

样例输出4：

121

**Data Constraint**

**Hint**

【数据规模】  
数据 1 2~3 4~6 7~12 13~19 20~25  
n 3 4 5 6 7 8

**1667. 【AHOI2009】中国象棋 (Standard IO)**

Time Limits: **1000 ms**  Memory Limits: **65536 KB**  Detailed Limits

**Description**

　　在N行M列的棋盘上，放若干个炮可以是0个，使得没有任何一个炮可以攻击另一个炮。请问有多少种放置方法？中国象棋中炮的行走方式大家应该很清楚吧.

**Input**

　　一行包含两个整数N，M，中间用空格分开.

**Output**

　　输出所有的方案数，由于值比较大，输出其mod 9999973

**Sample Input**

1 3

**Sample Output**

7

**Data Constraint**

**Hint**

　　除了在3个格子中都放满炮的的情况外，其它的都可以.   
　　100%的数据中N,M不超过100   
　　50%的数据中，N,M至少有一个数不超过8   
　　30%的数据中，N,M均不超过6

**1663. 【AHOI2009】维护序列 (Standard IO)**

Time Limits: **3000 ms**  Memory Limits: **128000 KB**  Detailed Limits

**Description**

　　老师交给小可可一个维护数列的任务，现在小可可希望你来帮他完成。  
　　有长为N的数列，不妨设为a1,a2,…,aN 。有如下三种操作形式：  
　　(1)把数列中的一段数全部乘一个值;  
　　(2)把数列中的一段数全部加一个值;  
　　(3)询问数列中的一段数的和，由于答案可能很大，你只需输出这个数模P的值。

**Input**

　　第一行两个整数N和P(1≤P≤1000000000）。第二行含有N个非负整数,从左到右依次为a1,a2,…,aN, (0≤ai≤1000000000,1≤i≤N)。第三行有一个整数M，表示操作总数。从第四行开始每行描述一个操作，输入的操作有以下三种形式：  
　　操作1：“1 t g c”(不含双引号)。表示把所有满足t≤i≤g的ai改为ai×c (1≤t≤g≤N,0≤c≤1000000000)。  
　　操作2：“2 t g c”(不含双引号)。表示把所有满足t≤i≤g的ai改为ai+c (1≤t≤g≤N,0≤c≤1000000000)。  
　　操作3：“3 t g”(不含双引号)。询问所有满足t≤i≤g的ai的和模P的值  
　　(1≤t≤g≤N)。  
　　同一行相邻两数之间用一个空格隔开，每行开头和末尾没有多余空格。

**Output**

　　对每个操作3，按照它在输入中出现的顺序，依次输出一行一个整数表示询问结果。

**Sample Input**

7 43

1 2 3 4 5 6 7

5

1 2 5 5

3 2 4

2 3 7 9

3 1 3

3 4 7

**Sample Output**

2

35

8

**Data Constraint**

**Hint**

【样例说明】  
　　初始时数列为(1,2,3,4,5,6,7)。  
　　经过第1次操作后，数列为(1,10,15,20,25,6,7)。  
　　对第2次操作，和为10+15+20=45，模43的结果是2。  
　　经过第3次操作后，数列为(1,10,24,29,34,15,16}  
　　对第4次操作，和为1+10+24=35，模43的结果是35。  
　　对第5次操作，和为29+34+15+16=94,模43的结果是8。

【数据规模和约定】  
　　测试数据规模如下表所示:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据编号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| N= | 10 | 1000 | 1000 | 10000 | 60000 | 70000 | 80000 | 90000 | 100000 | 100000 |
| M= | 10 | 1000 | 1000 | 10000 | 60000 | 70000 | 80000 | 90000 | 100000 | 100000 |