ioi2016 中国国家集训队第一次作业：试题泛做

长沙市长郡中学 任瀚林

2015 年 12 月 27 日

完成情况：102道(84×hard+18×challenge)，其中两道(QRECT和TREECNT2)在清橙上被卡常数。

注意：对于我加了卡时的challenge型试题，“时间复杂度”一栏均为$O(init + count \times update)$，其中$init$是初始化复杂度，$update$是一次调整/形成新解的复杂度。

字符串相关的题目，σ代表字符集大小。

目 录

1 GNUM **12**
1.1 题目描述 . . . 12
1.2 数据范围 . . . 12
1.3 解题报告 . . . 12

2 SEAEQ **12**
2.1 题目描述 . . . 12
2.2 数据范围 . . . 12
2.3 解题报告 . . . 12

3 PRIMEDST **13**
3.1 题目描述 . . . 13
3.2 数据范围 . . . 13
3.3 解题报告 . . . 13

4 DGCD **13**
4.1 题目描述 . . . 13
4.2 数据范围 . . . 13
4.3 解题报告 . . . 14

5 LYRC **14**
5.1 题目描述 . . . 14
5.2 数据范围 . . . 14
5.3 解题报告 . . . 14

6 GERALD07 **14**
6.1 题目描述 . . . 14
6.2 数据范围 . . . 14
6.3 解题报告 . . . 14

7 TAPAIR **15**
7.1 题目描述 . . . 15
7.2 数据范围 . . . 15
7.3 解题报告 . . . 15

8 XRQRS **15**
8.1 题目描述 . . . 15
8.2 数据范围 . . . 15
8.3 解题报告 . . . 15

9 QUERY **16**
9.1 题目描述 . . . 16
9.2 数据范围 . . . 16
9.3 解题报告 . . . 16

10 FIBTREE **16**
10.1 题目描述 . . . 16
10.2 数据范围 . . . 16
10.3 解题报告 . . . 16

11 TSUBSTR **17**
11.1 题目描述 . . . 17
11.2 数据范围 . . . 17
11.3 解题报告 . . . 17

12 COUNTARI **17**
12.1 题目描述 . . . 17
12.2 数据范围 . . . 17
12.3 解题报告 . . . 17

13 TWOCOMP **18**
13.1 题目描述 . . . 18
13.2 数据范围 . . . 18
13.3 解题报告 . . . 18

14 CARDSURF **18**
14.1 题目描述 . . . 18
14.2 数据范围 . . . 18
14.3 解题报告 . . . 19

15 MONOPLOY **19**
15.1 题目描述 . . . 19
15.2 数据范围 . . . 19
15.3 解题报告 . . . 19

16 FNCS **19**
16.1 题目描述 . . . 19
16.2 数据范围 . . . 19
16.3 解题报告 . . . 19

17 DIFTRIP 20
17.1 题目描述 . . . 20
17.2 数据范围 . . . 20
17.3 解题报告 . . . 20

18 FN 20
18.1 题目描述 . . . 20
18.2 数据范围 . . . 20
18.3 解题报告 . . . 20

19 RIN 21
19.1 题目描述 . . . 21
19.2 数据范围 . . . 21
19.3 解题报告 . . . 21

20 BWGAME 21
20.1 题目描述 . . . 21
20.2 数据范围 . . . 22
20.3 解题报告 . . . 22

21 ANUDTQ 22
21.1 题目描述 . . . 22
21.2 数据范围 . . . 22
21.3 解题报告 . . . 22

22 CNTDSETS 22
22.1 题目描述 . . . 22
22.2 数据范围 . . . 23
22.3 解题报告 . . . 23

23 TREECNT2 23
23.1 题目描述 . . . 23
23.2 数据范围 . . . 23
23.3 解题报告 . . . 23

24 KNGHTMOV 24
24.1 题目描述 . . . 24
24.2 数据范围 . . . 24
24.3 解题报告 . . . 24

25 CBAL 24
25.1 题目描述 . . . 24
25.2 数据范围 . . . 25
25.3 解题报告 . . . 25

26 QTREE 25
26.1 题目描述 . . . 25
26.2 数据范围 . . . 25
26.3 解题报告 . . . 25

27 challenge GERALD09 26
27.1 题目描述 . . . 26
27.2 数据范围 . . . 26
27.3 解题报告 . . . 26

28 BAKE 26
28.1 题目描述 . . . 26
28.2 数据范围 . . . 27
28.3 解题报告 . . . 27

29 ANDOOR 27
29.1 题目描述 . . . 27
29.2 数据范围 . . . 27
29.3 解题报告 . . . 27

30 PARADE 27
30.1 题目描述 . . . 27
30.2 数据范围 . . . 27
30.3 解题报告 . . . 27

31 MATCH 28
31.1 题目描述 . . . 28
31.2 数据范围 . . . 28
31.3 解题报告 . . . 28

32 QRECT 28
32.1 题目描述 . . . 28
32.2 数据范围 . . . 29
32.3 解题报告 . . . 29

33 TRIPS 29
33.1 题目描述 . . . 29
33.2 数据范围 . . . 29
33.3 解题报告 . . . 29

34 CLOWAY 30
34.1 题目描述 . . . 30
34.2 数据范围 . . . 30
34.3 解题报告 . . . 30

35 EASYEX 30
35.1 题目描述 . . . 30
35.2 数据范围 . . . 31
35.3 解题报告 . . . 31

36 RNG 31
36.1 题目描述 . . . 31
36.2 数据范围 . . . 31
36.3 解题报告 . . . 32

37 HAMILG **32**
37.1 题目描述 32
37.2 数据范围 32
37.3 解题报告 33

38 GTHRONES **33**
38.1 题目描述 33
38.2 数据范围 33
38.3 解题报告 33

39 CONPOIN **34**
39.1 题目描述 34
39.2 数据范围 34
39.3 解题报告 34

40 DIVIDEN **34**
40.1 题目描述 34
40.2 数据范围 34
40.3 解题报告 34

41 STREETTA **35**
41.1 题目描述 35
41.2 数据范围 35
41.3 解题报告 35

42 challenge CLOSEST **35**
42.1 题目描述 35
42.2 数据范围 35
42.3 解题报告 35

43 RANKA **36**
43.1 题目描述 36
43.2 数据范围 36
43.3 解题报告 36

44 SEAARC **36**
44.1 题目描述 36
44.2 数据范围 36
44.3 解题报告 36

45 LPARTY **37**
45.1 题目描述 37
45.2 数据范围 37
45.3 解题报告 37

46 SEINC **38**
46.1 题目描述 38
46.2 数据范围 38
46.3 解题报告 38

47 TKCONVEX 38
47.1 题目描述 . . . 38
47.2 数据范围 . . . 38
47.3 解题报告 . . . 38

48 TWOROADS 39
48.1 题目描述 . . . 39
48.2 数据范围 . . . 39
48.3 解题报告 . . . 39

49 DEVLOCK 39
49.1 题目描述 . . . 39
49.2 数据范围 . . . 39
49.3 解题报告 . . . 40

50 MISINT2 40
50.1 题目描述 . . . 40
50.2 数据范围 . . . 40
50.3 解题报告 . . . 40

51 DAGCH 41
51.1 题目描述 . . . 41
51.2 数据范围 . . . 41
51.3 解题报告 . . . 41

52 GRAPHCNT 41
52.1 题目描述 . . . 41
52.2 数据范围 . . . 41
52.3 解题报告 . . . 42

53 CLONES 42
53.1 题目描述 . . . 42
53.2 数据范围 . . . 42
53.3 解题报告 . . . 42

54 LEBOXES 43
54.1 题目描述 . . . 43
54.2 数据范围 . . . 43
54.3 解题报告 . . . 43

55 QTREE6 43
55.1 题目描述 . . . 43
55.2 数据范围 . . . 43
55.3 解题报告 . . . 44

56 challenge COMPLEXT 44
56.1 题目描述 . . . 44
56.2 数据范围 . . . 44
56.3 解题报告 . . . 44

57 COT5 44
57.1 题目描述 44
57.2 数据范围 44
57.3 解题报告 44

58 FINDSEQ 45
58.1 题目描述 45
58.2 数据范围 45
58.3 解题报告 45

59 CONNECT 45
59.1 题目描述 45
59.2 数据范围 45
59.3 解题报告 46

60 SHORT2 46
60.1 题目描述 46
60.2 数据范围 46
60.3 解题报告 46

61 SEAORD 47
61.1 题目描述 47
61.2 数据范围 47
61.3 解题报告 47

62 RIVPILE 47
62.1 题目描述 47
62.2 数据范围 47
62.3 解题报告 47

63 CUCUMBER 48
63.1 题目描述 48
63.2 数据范围 48
63.3 解题报告 48

64 REALSET 48
64.1 题目描述 48
64.2 数据范围 48
64.3 解题报告 49

65 CUSTPRIM 49
65.1 题目描述 49
65.2 数据范围 49
65.3 解题报告 49

66 SPMATRIX 50
66.1 题目描述 50
66.2 数据范围 50
66.3 解题报告 50

67 MINESREV 50
67.1 题目描述 . . . 50
67.2 数据范围 . . . 50
67.3 解题报告 . . . 51

68 TICKETS 51
68.1 题目描述 . . . 51
68.2 数据范围 . . . 51
68.3 解题报告 . . . 51

69 QPOINT 52
69.1 题目描述 . . . 52
69.2 数据范围 . . . 52
69.3 解题报告 . . . 52

70 PUSHFLOW 52
70.1 题目描述 . . . 52
70.2 数据范围 . . . 52
70.3 解题报告 . . . 53

71 challenge SEAVEC 53
71.1 题目描述 . . . 53
71.2 数据范围 . . . 53
71.3 解题报告 . . . 53

72 PARSIN 53
72.1 题目描述 . . . 53
72.2 数据范围 . . . 54
72.3 解题报告 . . . 54

73 SHORTCIR 54
73.1 题目描述 . . . 54
73.2 数据范围 . . . 54
73.3 解题报告 . . . 54

74 challenge SIMGRAPH 55
74.1 题目描述 . . . 55
74.2 数据范围 . . . 55
74.3 解题报告 . . . 55

75 challenge FAULT 55
75.1 题目描述 . . . 55
75.2 数据范围 . . . 55
75.3 解题报告 . . . 55

76 FLYDIST 55
76.1 题目描述 . . . 55
76.2 数据范围 . . . 56
76.3 解题报告 . . . 56

77 challenge SPELL 56
77.1 题目描述 . . . 56
77.2 数据范围 . . . 56
77.3 解题报告 . . . 56

78 LUCKYDAY 57
78.1 题目描述 . . . 57
78.2 数据范围 . . . 57
78.3 解题报告 . . . 57

79 BB 57
79.1 题目描述 . . . 57
79.2 数据范围 . . . 57
79.3 解题报告 . . . 57

80 challenge BICKER 58
80.1 题目描述 . . . 58
80.2 数据范围 . . . 58
80.3 解题报告 . . . 58

81 challenge CKROACH 58
81.1 题目描述 . . . 58
81.2 数据范围 . . . 58
81.3 解题报告 . . . 58

82 challenge SEAND2 59
82.1 题目描述 . . . 59
82.2 数据范围 . . . 59
82.3 解题报告 . . . 59

83 challenge MAXRECT 59
83.1 题目描述 . . . 59
83.2 数据范围 . . . 59
83.3 解题报告 . . . 59

84 GERALD08 59
84.1 题目描述 . . . 59
84.2 数据范围 . . . 60
84.3 解题报告 . . . 60

85 LECOINS 60
85.1 题目描述 . . . 60
85.2 数据范围 . . . 60
85.3 解题报告 . . . 60

86 EVILBOOK 61
86.1 题目描述 . . . 61
86.2 数据范围 . . . 61
86.3 解题报告 . . . 61

87 CHEFBOOK 62
87.1 题目描述 . . . 62
87.2 数据范围 . . . 62
87.3 解题报告 . . . 62

88 challenge SIMNIM 63
88.1 题目描述 . . . 63
88.2 数据范围 . . . 63
88.3 解题报告 . . . 63

89 LMATRIX3 63
89.1 题目描述 . . . 63
89.2 数据范围 . . . 63
89.3 解题报告 . . . 63

90 challenge WORDNINJ 64
90.1 题目描述 . . . 64
90.2 数据范围 . . . 65
90.3 解题报告 . . . 65

91 challenge MANYLEFT 65
91.1 题目描述 . . . 65
91.2 数据范围 . . . 65
91.3 解题报告 . . . 65

92 MAXCIR 66
92.1 题目描述 . . . 66
92.2 数据范围 . . . 66
92.3 解题报告 . . . 66

93 challenge CHAORNOT 66
93.1 题目描述 . . . 66
93.2 数据范围 . . . 66
93.3 解题报告 . . . 67

94 challenge MAXDIFFW 67
94.1 题目描述 . . . 67
94.2 数据范围 . . . 67
94.3 解题报告 . . . 67

95 COOLNUM 68
95.1 题目描述 . . . 68
95.2 数据范围 . . . 68
95.3 解题报告 . . . 68

96 SHORT 68
96.1 题目描述 . . . 68
96.2 数据范围 . . . 68
96.3 解题报告 . . . 68

97 MARTARTS **69**

97.1 题目描述 69

97.2 数据范围 69

97.3 解题报告 69

98 CNTHEX **69**

98.1 题目描述 69

98.2 数据范围 69

98.3 解题报告 70

99 challenge LAND **70**

99.1 题目描述 70

99.2 数据范围 70

99.3 解题报告 70

100 SIGFIB **70**

100.1 题目描述 70

100.2 数据范围 70

100.3 解题报告 71

101 CIELQUAK **71**

101.1 题目描述 71

101.2 数据范围 71

101.3 解题报告 72

102 challenge SEAPERM **72**

102.1 题目描述 72

102.2 数据范围 72

102.3 解题报告 72

§1 GNUM

§1.1 题目描述

有两个长度为N的数组A,B和两个二元组的集合S_1,S_2，初始时$S_1=S_2=\emptyset$。现在有一种操作：

每次选两个二元组$(i,j),(p,q)$，满足$(i,j)\notin S_1,(p,q)\notin S_2,B_j>A_i,B_p<A_q,\gcd(A_i,B_j,A_q,B_p)>1$，如果这两个二元组存在就把$(i,j)$加入$S_1$，把$(p,q)$加入$S_2$。

问最多进行多少次操作。

§1.2 数据范围

数据组数不超过10组，$N\le 400,A_i,B_i\le 10^9$。

§1.3 解题报告

很明显答案要求某个图的最大匹配——考虑一个二分图，左边的点代表所有满足$B_j>A_i,\gcd(A_i,B_j)>1$的二元组(i,j)，右边的点代表所有满足$B_p<A_q,\gcd(A_q,B_p)>1$的二元组(p,q)。如果$\gcd(A_i,B_j,A_q,B_p)>1$那么给二元组(i,j)和(p,q)连边。

用U表示左边所有点的$\gcd(A_i,B_j)$组成的数组，V表示右边所有点的$\gcd(A_q,B_p)$组成的数组。最大匹配可以用Dinic算法求：源点向所有U_i连边，可以连边的U_i向V_j连边，V_j向T连边。

考虑优化连边，对每一个可能出现的质数p，如果$p|U_i$那么U_i向p连边；如果$p|V_j$那么p向V_j连边。这样连边的复杂度是$O(XN^2)$(一个数至多有$X=9$个不同的质因子)。

然后把相等的U合并，把相等的V合并即可通过此题。

时间复杂度：$O(\mathrm{networkflow}(N^2,XN^2))$(可能达不到，因为合并相同数之后程序不好卡)

空间复杂度：$O(XN^2)$

§2 SEAEQ

§2.1 题目描述

称两个长度为n的数组A,B相似，如果对于所有$i(1\le i\le n)$，满足$C(A,A_i)=C(B,B_i)$。其中$C(X,x)$等于满足$X[j]<x(1\le j\le n)$的j的数目。

对于两个排列P_1,P_2，定义函数$F(P_1,P_2)$等于满足$P_1[l\dots r]$相似于$P_2[l\dots r](1\le l\le r\le n)$并且$P_1[l\dots r]$包含不超过$E$个逆序对的数对$(l,r)$的数目。

求对P_1,P_2取遍所有n个元素的排列$F(P_1,P_2)$的总和是多少。

§2.2 数据范围

数据组数不超过10^4组，$n\le 500,E\le 10^6$。

§2.3 解题报告

令$Perm(n)$表示$1\dots n$的排列的全集，$inv(p)$表示排列p的逆序对数，$p[l\dots r]$表示序列p的第l项到第r项之间的元素依次组成的新序列。对于一个没有重复元素的序列p，$f(p)$表示一个$1\dots n$的排列满足$f(p)_i=C(p,p_i)+1$。

我们要求的便是

$$
\begin{aligned}
\text{要求} &= \sum_{P_1 \in Perm(n)} \sum_{P_2 \in Perm(n)} \sum_{1 \le l \le r \le n} [f(P_1[l \ldots r]) = f(P_2[l \ldots r])][inv(f(P_1[l \ldots r])) \le E] \\
&= \sum_{1 \le l \le r \le n} \sum_{p \in Perm(r-l+1)} [inv(p) \le E] \left(\sum_{P_1 \in Perm(n)} [f(P_1[l \ldots r]) = p] \right)^2 \\
&= \sum_{x=1}^{n} (n-x+1) \sum_{p \in Perm(x)} [inv(p) \le E] \left(\frac{n!}{x!} \right)^2 \\
&= \sum_{x=1}^{n} (n-x+1) \left(\sum_{i=0}^{E} cnt_{x,i} \right) \left(\frac{n!}{x!} \right)^2
\end{aligned}
$$

其中$cnt_{x,E}$表示长度为x的排列中，逆序对数为E的排列个数。

注意到这个cnt数组显然是可以dp+前缀和优化的。具体地说，就是记$Scnt_{x,E} = \sum_{i=1}^{E} cnt_{x,i}$。在求完$cnt_{x-1}$之后，可以用$O(E)$的时间求得$Scnt_{x-1}$，而

$$
cnt_{x,E} = \begin{cases} Scnt_{x-1,E} - Scnt_{x-1,E-n} & E \ge n \\ Scnt_{x-1,E} & E < n \end{cases}
$$

所以用$O(n^3)$的时间求出$Scnt$数组之后，可以$O(n)$地回答询问。

时间复杂度：$O(n^3 + Tn)$

空间复杂度：$O(n^3)$

§3 PRIMEDST

§3.1 题目描述

给出一棵n个节点的树，求随机选两个不同点，它们距离是质数的概率。

§3.2 数据范围

$2 \le n \le 50000$。

§3.3 解题报告

求出质数距离点对的个数除以$\binom{n}{2}$就可以了。

考虑边分治。处理一条边(u, v)的时候，记a_i为u的子树中到u的距离为i的点数，b_j为v的子树中到v的距离为j的点数(当然不包括边分治时加的虚点)。

问题变成了$\sum_{i+j+w_{u,v} \in Primes} a_i b_j$，FFT即可。其中因为边分治要加虚点所以$w_{u,v} \in \{0, 1\}$。

时间复杂度：$O(n \log^2 n)$

空间复杂度：$O(n)$

§4 DGCD

§4.1 题目描述

给出一棵n个节点的树，每个点有点权v_i(正整数)，支持两种操作共Q个：

- 给一条路径上的点权值加上一个整数d;
- 询问一条路径上的点权值的gcd。

§4.2 数据范围

$1 \le n, Q \le 50000, 1 \le v_i \le 10^4, 0 \le d \le 10^4$。

§4.3 解题报告

考虑树链剖分，对于每个点维护u_i表示它的权值减去它重儿子的权值(叶子节点$u_i = v_i$)，同时也维护v_i表示它的当前权值。

修改(x, y)的时候对v_i进行链加法，这个可以用一棵线段树维护。然后考虑维护u值，只有这么些点的u值可能会变：访问到的每个轻边的父亲节点；x；y；$\text{lca}(x, y)$的父亲。一共$O(\log n)$个，线段树维护，单点修改，区间取gcd即可。

询问的时候考虑拆成若干段重链$p \to c$(p是祖先)各自询问，然后最后把答案合起来。重链的答案就是c的父亲到p这一段的u值的gcd再与v_c取gcd。

时间复杂度：$O(n + q\log^3 n)$

空间复杂度：$O(n)$

§5 LYRC

§5.1 题目描述

给出W个单词和N个字符串，询问每个单词在所有字符串中出现的次数之和。

§5.2 数据范围

$1 \le W \le 500$，$1 \le |P|$(单词长度)≤ 5000，$1 \le N \le 100$，$1 \le |S|$(字符串长度)≤ 10000。字符集大小为63。

§5.3 解题报告

首先对所有单词建出AC自动机，然后把所有字符串放在自动机上跑。跑到一个节点的时候给它及它$fail$链上的所有祖先答案+1。然后对每个单词看它对应的节点答案是多少即可。

当然这样是会TLE的。我们改成跑到一个节点时只给它自己+1。我们发现$fail$链组成了一个树结构，对每个单词节点查询子树和就是它的答案。子树和直接dp即可。

时间复杂度：$O(\sigma W|P| + N|S|)$

空间复杂度：$O(\sigma W|P|)$

§6 GERALD07

§6.1 题目描述

给出一个N个点M条边的无向图G和Q对询问L_i, R_i，回答仅保留编号在$[L_i, R_i]$内的边时，G的连通块个数。

§6.2 数据范围

数据组数不超过2000，N, M, Q的总和不超过200000。

§6.3 解题报告

对一条编号为i的边，记录f_i表示最小的l，使得编号在$[l, i-1]$内的边无法连通i的两端。

f_i的求法如下：用LCT维护关于编号的最大生成树，插入一条边$i = (u, v)$的时候，如果u, v不连通那么$f_i = 1$；否则f_i是u到v路径上编号最小的边e的编号加1。然后用i替换e即可。

对于一次询问$[L, R]$，每有一条边i满足$L \le i \le R, f_i \le L$，图G的连通块个数就会减1。故只要求出满足$L \le i \le R, f_i \le L$的边数即可。这个是可以用主席树进行维护的。

时间复杂度：$O(M\log N + Q\log M)$

空间复杂度：$O(M\log M)$

§7 TAPAIR

§7.1 题目描述

给出一张N个点M条边的无向简单图，求有多少**边的无序二元组**，满足删掉这两条边之后图仍然连通。

§7.2 数据范围

$1 \le N \le 100000, 1 \le M \le 300000$。

§7.3 解题报告

对这张图进行dfs，给每一条非树边赋一个$[1, 2^{64} - 1]$内的随机权值，每一条树边的权值是覆盖它的非树边权值的异或和。考虑一个点，很容易证明它的所有邻边的权值异或和为0。这样推广一下，会发现图的任意一个割的权值异或和都是0。

这样，如果删除一个边集之后图不连通，那么这个边集一定有一个非空子集，权值异或和为0。而一个大小为2的边集存在一个权值异或和为0的非空子集但是删除之后图仍然连通的概率很小。

所以相当于给定M个数(边权)$W_1, W_2, \ldots, W_M$，问多少个无序对(i, j)满足$W_i \neq 0, W_j \neq 0, W_i \neq W_j$。剔除所有0，对每一个不同的$W_i$开一个Hash统计其出现次数即可。

那么，如何在$O(M)$的时间内求出树边的权值呢?假设dfs到点x，我们确定x到所有儿子的边的权值的异或和，与所有连向x的非树边权值异或一下就是x到其父亲的边的权值。

时间复杂度：$O(M)$

空间复杂度：$O(M)$

§8 XRQRS

§8.1 题目描述

维护一个初始为空的序列，支持如下操作：

- $0\ x$：在序列最后加入数字x；
- $1\ L\ R\ x$：在区间$[L \ldots R]$中找到数字y，最大化$x \text{ xor } y$；
- $2\ k$：删除最后k个元素；
- $3\ L\ R\ x$：求区间$[L \ldots R]$中，小于等于x的数的个数；
- $4\ L\ R\ k$：求区间$[L \ldots R]$的第k小的数。

§8.2 数据范围

$M, x \le 5 \times 10^5$。

§8.3 解题报告

考虑对每一个$i \le$当前序列长度，维护一棵Trie T_i代表$1 \ldots i$内所有数。使用可持久化Trie可以轻松维护。

然后更新这些Trie也是很容易的。插入的时候：设N为插入前的序列长度，$T_{N+1} = T_N$，然后往T_{N+1}内插入x即可。删除的时候直接移动序列的尾指针即可。

如果要提取一个区间$[L \ldots R]$的Trie的话，可以直接进行区间减法，即$T_R - T_{L-1}$。我们并不需要建出这棵Trie，只需要知道它的特定节点的值即可。

接下来的问题就是：给出一个Trie，如何$O(\log x)$地查询Trie中：

- 与x异或最大的数：贪心地往反方向走即可。
- 小于等于x的数的个数：在Trie中沿着$x+1$走，将所有左子树的大小相加。
- 第k小的数：在Trie上二分答案。

时间复杂度：$O(M\log x)$

空间复杂度：$O(M\log x)$

§9 QUERY

§9.1 题目描述

给定一棵n个节点的树，每个点有权值，初始为0。维护m个操作：

- 1 x y a b：x到y的路径上的点权值增加，第一个点(x)增加a，第二个点增加$a+b$，第三个点增加$a+2b$，以此类推。
- 2 x y：求x到y路径上的点权和。
- 3 x：把树的状态变成第x次操作后的状态。

强制在线。

§9.2 数据范围

$1\le n,m\le 10^5, 1\le a,b\le 1000$。

§9.3 解题报告

直接对树进行树链剖分，树链剖分之后用线段树维护dfs序。

对线段树的每个节点维护一个标记(a,b)，表示该节点对应的区间的第$i(0\le i< size)$个数加$a+bi$。标记可以很好地合并、下传，区间和也可以很容易地更新。

把线段树可持久化就可以通过本题了。注意数据有点卡，线段树的节点数需要开到3×10^7。

时间复杂度：$O(n+m\log^2 n)$

空间复杂度：$O(n+m\log^2 n)$

§10 FIBTREE

§10.1 题目描述

斐波那契数列是指这样一个数列：$f_1=f_2=1, f_i=f_{i-1}+f_{i-2}(i>2)$。给出一棵$N$个节点的有根树，在线处理$M$个询问：

- A x y：将x到y路径上第k个节点的权值加上f_k。(第1个节点是x)
- QS x y：求以x为根时y的子树和。
- QC x y：求x到y路径上的点权和。
- R x：将所有节点权值还原到第x个询问后的状态。

§10.2 数据范围

$1\le N,M\le 10^5$。

§10.3 解题报告

考虑一个可持久化的数组T，支持$O(H(N))$询问区间和，以及下面两个操作：

- ADD1 l r a b：令$f_1=a, f_2=b, f_i=f_{i-1}+f_{i-2}(i>2)$，实行$T_{l+i-1}\leftarrow T_{l+i-1}+f_i(l\le l+i-1\le r)$；
- ADD2 l r a b：令$f_1=a, f_2=b, f_i=f_{i-1}+f_{i-2}(i>2)$，实行$T_{r-i+1}\leftarrow T_{r-i+1}+f_i(l\le r-i+1\le r)$；

我们用这个数组维护树链剖分的dfs序，就可以用$O(H(N)\log N)$的时间完成题面中的所有操作。

现在考虑这个数组的实现，用可持久化线段树，每个点维护两个标记：一个是ADD1操作的标记，一个是ADD2操作的标记。

给定f_1, f_2, n可以$O(1)$求出$\sum_{i=1}^{n} f_i$，那么就可以快速打标记；可以$O(1)$求出f_i那么就可以下传标记。而这些通过预处理Fibonacci的转移矩阵是可以很快算出的。并且本题标记支持合并，所以线段树是可以做的。

时间复杂度：$O(N + M\log^2 N)$

空间复杂度：$O(N + M\log^2 N)$

§11 TSUBSTR

§11.1 题目描述

给出一个N个节点的Trie，定义其**子串**为从Trie中某个节点走向其子孙所经过的路径上的字符连起来得到的串。求该Trie的不同子串数。同时，你需要解决Q个询问，每个询问给出26个字母的一个排列作为大小顺序，求字典序第K_i小的子串。

§11.2 数据范围

$1 \le N \le 250000, 1 \le Q \le 50000, 1 \le K_i < 2^{63}$，答案不超过800KB，Trie上的字符随机生成。

§11.3 解题报告

考虑建出该Trie的后缀自动机(SAM，见2015集训队论文)，令$go_{x,y}$表示SAM上点x的出边中，写着字母y的出边指向的点。利用经典的SAM上dp即可求出第一问。即，令f_i表示从SAM的起点S走到点i的方案数，那么$f_S = 1$，f_i对所有$f_{go_{i,j}}$有贡献。最后将f值求和即可得到答案。

关于第二问：令$F(x, K)$表示从x出发走到某个接受态的所有方案(子串)中，字典序第K大的那个；令g_x表示从x出发走一条路径的方案数。那么$g_x = \sum_{c\in\sigma} g_{go_{x,c}} + 1$。考虑$F(x, K)$，首先$K = 1$时显然答案是空串；否则$K$减一，然后按顺序枚举第1个字符是谁(设为$c$)，如果$g_{go_{i,c}} < K$那么$K \leftarrow K - g_{go_{i,c}}$，枚举下一个字符；否则问题变成$F(go_{i,c}, K)$。每走一步会输出一个字符，所以复杂度是与输出量(设为O)乘σ同级的。

时间复杂度：$O(\sigma(O + N))$

空间复杂度：$O(\sigma N)$

§12 COUNTARI

§12.1 题目描述

给定一个长度为N的数列A，求三元组$(i, j, k)(1 \le i < j < k \le N)$的数目，使得$A_i + A_k = 2A_j$。

§12.2 数据范围

$1 \le N \le 10^5, 1 \le A_i \le 30000$。

§12.3 解题报告

令a为A数组中数的最大值。

枚举j。我们需要求出满足$i < j, k > j, A_i + A_k = 2A_j$的数对$(i, k)$个数。

我们令l_i为$A_1, A_2, \ldots, A_{j-1}$中等于$i$的数的个数；$r_i$为$A_{j+1}, A_{j+2}, \ldots, A_n$中等于$i$的数的个数。答案就是$l, r$的卷积的第$2A_j$项。

我们考虑用定期重建的方法维护卷积。把序列每$O(f(a))$个元素分一块，每扫到一个块尾的时候重新计算l, r两个数组及其卷积；询问卷积第$2A_j$项的时候，我们扫描所有的修改，每次修改可以$O(1)$地更新l, r数组和答案。最后把l, r数组还原即可。

询问复杂度$O(f(a))$，重建复杂度$O(\frac{n}{f(a)} a\log a)$

令$O(nf(a)) = O(\frac{na\log a}{f(a)})$，得$f(a) = O(\sqrt{a\log a})$。考虑到FFT常数较大，$f(a)$可稍微取大。

注意到$l+r+x^{A_j}$就是总的序列，所以我们可以先预处理出总序列的DFT值，重建的时候只需要两次DFT，常数更小。(这题还是挺卡常数的)

时间复杂度：$O(n\sqrt{a\log a})$

空间复杂度：$O(n+a)$

§13 TWOCOMP

§13.1 题目描述

给定一棵N个节点的树，两个公司BTS和GTN各有一些线路(树上某两点之间的路径)。BTS有M_1条线路，GTN有M_2条线路。请选择一些线路使得不同公司的线路不相交，且线路的权值之和最大。

§13.2 数据范围

$1\le N\le 10^5, 1\le M_1,M_2\le 700$，权值不超过$10^6$。

§13.3 解题报告

首先用$O(1)$时间解决路径相交问题：给出树上两条路径(x,y)和(u,v)，求它们是否相交。考虑$z=\mathrm{lca}(x,y), w=\mathrm{lca}(u,v)$，我们知道LCA是可以$O(N\log N)-O(1)$用RMQ求的，那么我们只需要考虑$((x,z),(u,w)), ((x,z),(v,w)), ((y,z),(u,w)), ((y,z),(v,w))$这四对路径是否相交即可。现在我们把问题变成了$(p_1,c_1)$与$(p_2,c_2)$是否相交，其中$p_1,p_2$分别是$c_1,c_2$的祖先。而相交的充要条件是$p_1,c_1$中存在点在路径$(p_2,c_2)$上，或$p_2,c_2$中存在点在路径$(p_1,c_1)$上。由于是祖孙关系的路径，所以判断点是否在路径上只需要判断两次子树关系即可。而得到dfs序之后，子树关系是可以$O(1)$求的。

现在考虑每一条路径是一个点，好像就是这么个问题：**二分图最大点权独立集**。考虑如下的建图：S向所有X部点连边，权值为点权；所有X部点向与其相连的Y部点连边，权值为$+\infty$；所有Y部点向T连边，权值为点权。在这个图中，我们令割掉一条边表示不选这条边相邻的点(一定会割掉与S或T相邻的边，因为其他边权值为$+\infty$。所以“相邻点”就是指原来二分图中的点)。考虑有边相连的$S\to x\to y\to T$，最小割的定义“$S\to x$与$y\to T$至少被割掉一个”与独立集的定义“x,y不能同时都选”正好吻合!

对图跑一遍最小割就是答案。

时间复杂度：$O(N\log N+\mathrm{networkflow}(M_1+M_2,M_1M_2))$

空间复杂度：$O(N\log N+M_1M_2)$

§14 CARDSURF

§14.1 题目描述

桌面上有n张牌排成一个栈，自顶向下编号为$1,2,\ldots,n$。每次给出三个整数a,b,c，执行如下操作：

1.把栈顶a张牌拿走；

2.把栈顶b张牌拿走；

3.把第1步中的a张牌按照原顺序放入栈中；

4.把栈顶c张牌拿走；

5.把第2步中的b张牌按照翻转顺序放入栈中；

6.把第4步中的c张牌按照原顺序放入栈中。

给出M次操作，求最后的牌顺序。

§14.2 数据范围

$n,M\le 10^5$，数据随机生成。

§14.3 解题报告

直接使用splay打翻转标记。

时间复杂度：$O((M+n)\log n)$

空间复杂度：$O(n)$

§15 MONOPLOY

§15.1 题目描述

给出一棵N个点的有根树，每个点有一个颜色，一条边的两端点颜色不同那么这条边的权值是1；否则这条边的权值是0。令$f(x)$为点x到根的最短路的权值。支持两种操作共Q个：

- O u：把点u到根的路径上所有点涂成一个从未出现过的颜色。
- q u：询问点u子树内所有点x的$f(x)$值的平均值。

§15.2 数据范围

数据组数不超过15，$N,Q\le 10^5$，所有数据中N,Q的总和不超过2×10^5。

§15.3 解题报告

考虑将权值为0的边称作"重边"，权值为1的边称作"轻边"，我们发现操作O的意思就是将u连向所有儿子的边置为轻边，将u到祖先经过的所有边置为重边。这正是LCT中的Access操作。

我们用LCT维护这棵树，每次执行O操作的时候直接Access，复杂度就有保证了。现在考虑q操作，我们考虑直接维护$f(x)$，考虑改变一条边的权值时f值的变化情况，这条边的儿子节点的子树的f值加或减了1。我们用线段树维护树的dfs序，改变一条边的时候子树加减变成区间加减，询问的时候子树求和变成区间求和。

时间复杂度：$O((N+Q)\log^2 N)$

空间复杂度：$O(N)$

§16 FNCS

§16.1 题目描述

给出长度为N的数列A以及N个函数$f_1,f_2,\ldots,f_N$，第i个函数用两个正整数L_i,R_i表示，即$f_i=\sum_{p=L_i}^{R_i}A_p$。支持两种操作：

- 1 x y：赋值$A_x\leftarrow y$；
- 2 m n：求$\sum_{i=m}^{n}f_i$。

§16.2 数据范围

$1\le N\le 10^5, 1\le A_i,y\le 10^9, 1\le Q\le 10^5$。

§16.3 解题报告

考虑"单点修改，区间求和"的经典问题，我们给出一个修改复杂度$O(\sqrt{n})$，询问复杂度$O(1)$的算法。将序列每$O(\sqrt{n})$个元素分一块，每一块维护块内前缀和，另外维护p_i表示前i个块内所有元素的和。设修改块x内的元素，那么更新块x的块内前缀和，更新$p_x,p_{x+1},\ldots$。复杂度$O(\sqrt{n})$；询问某区间的和，转化为前缀和，可以看成一些块的前缀和加上剩下元素的和，可以$O(1)$询问。

现在考虑对函数分块，连续的$O(\sqrt{n})$个函数一块。对每一块i和序列元素j我们维护$f_{i,j}$为块i内包含j的函数个数；另外维护一个值表示当前块内所有函数值之和。另外我们用上面的分块方法维护序列。单点修改的时候使用$O(\sqrt{n})$的单点修改，并对所有函数块更新其函数值之和。查询$[m,n]$之间函数

的和的时候，如果一个块完全被$[m,n]$包含，那么直接拿出其函数值之和；否则，如果这个块有部分被$[m,n]$包含，由于这样的块只有$O(1)$个，所以暴扫块中的函数，对每一个函数做一次$O(1)$的区间查询。复杂度都是$O(\sqrt{n})$。

请注意本题需要使用`unsigned long long int`。

时间复杂度：$O((n+Q)\sqrt{n})$

空间复杂度：$O(n\sqrt{n})$

§17 DIFTRIP

§17.1 题目描述

考虑一棵有根树，“路径”被定义为从一个点到它的子树中的某点，依次经过的所有点组成的有序集。两个路径是等价的，当且仅当两条路径长度相等，且对于任意i，两条路径的第i个点在树中的度数相等。

求本质不同的路径条数。

§17.2 数据范围

$1 \le N \le 10^5$。

§17.3 解题报告

即Trie的本质不同的子串个数。

我们换一种定义，令一条“路径”为一个点到它的祖先依次经过的所有点组成的有序集。容易发现答案不变。此外我们令$s(x)$为点x到根所代表的路径。

我们给$s(x)$排序，然后统计所有点的深度之和减去相邻的$s(x)$的LCP(最长公共前缀)即可。排序可以用Trie上的后缀数组，LCP可以倍增求。设r_i为第i次基数排序之后的rank数组，$f_{i,j}$表示j的第2^i个祖先。那么求LCP(x,y)就是找到一个最大的i使得$r_{i,x}=r_{i,y}$，然后返回LCP$(f_{i,x},f_{i,y})+2^i$。利用倍增可以做到$O(\log N)$一次查询。

时间复杂度：$O(N\log N)$

空间复杂度：$O(N\log N)$

§18 FN

§18.1 题目描述

令

$$F_n=\begin{cases} n & 0\le n\le 1\\ F_{n-1}+F_{n-2} & n>1\end{cases}$$

求满足$F_n\equiv C(\mathrm{mod}\ P)$的最小$n$。如果$0\le n\le 2\times 10^9$的范围内没有找到所求的$n$，输出$-1$。

§18.2 数据范围

测试数据组数不超过100，$11\le P\le 2\times 10^9$，P是素数，$0\le C<P$，$P \bmod 10$是完全平方数。

§18.3 解题报告

根据二次互反律

$$\left(\frac{P}{5}\right)\left(\frac{5}{P}\right)=(-1)^{\frac{(P-1)(5-1)}{4}}=1$$

而$P \bmod 10$是完全平方数，$P>10$，P是质数，这意味着$P \bmod 5\in\{1,4\}$，即$\left(\frac{P}{5}\right)=1$。所以$\left(\frac{5}{P}\right)=1$，即5在模$P$意义下有二次剩余。

我们考虑开方：求$\sqrt{X} \bmod P$，令r为P的原根，BSGS求出$X = r^k$，如果k是偶数返回$r^{\frac{k}{2}}$；否则返回无解。此外，很容易改编BSGS算法，使得其返回最小的奇数/偶数解。

令$\alpha = \frac{1+\sqrt{5}}{2}, \beta = \frac{1-\sqrt{5}}{2}$(注意运算都是在模$P$意义下的)，注意到$\alpha\beta = -1$，有

$$C = F_n = \frac{1}{\sqrt{5}}(\alpha^n - \beta^n) = \frac{1}{\sqrt{5}}(\alpha^n - \frac{(-1)^n}{\alpha^n})$$

- n为偶数的时候，设$n = 2k$，那么

$$\sqrt{5}C = (\alpha^2)^k - \frac{1}{(\alpha^2)^k}$$

  是关于$(\alpha^2)^k$的一元二次方程，所以可以解出$(\alpha^2)^k$，通过BSGS可以解出k。

- n为奇数的时候，设$n = 2k+1$，那么

$$\sqrt{5}C = \alpha(\alpha^2)^k + \frac{1}{\alpha(\alpha^2)^k}$$

  也是关于$(\alpha^2)^k$的一元二次方程。同理用BSGS解出k。

无解的情况来自一元二次方程中$\sqrt{\Delta}$无解，或者根据奇偶性分类讨论时，解出的答案与原奇偶性不同。

时间复杂度：$O(T\sqrt{P})$

空间复杂度：$O(T\sqrt{P})$

§19 RIN

§19.1 题目描述

有N门课程，每门课程需要在M个学期中的某一个来完成。用$X[i][j]$表示第i门课程在第j个学期选修的收益，如果$X[i][j] = -1$则第i门课不能在第j个学期选修。

有K个条件：第A_i个课程选修完之后才能选修第B_i个课程。保证存在合理的课程选修方案。

求最大收益。

§19.2 数据范围

$1 \le M, N \le 100$，$0 \le K \le 100$，$-1 \le X[i][j] \le 100$。

§19.3 解题报告

其实这是一个很经典的最小割模型，类似于HNOI2013的切糕。

首先把问题变成最小化：每个$X[i][j]$变成$100 - X[i][j]$。如果$X[i][j] = -1$那么变成$+\infty$。下文中的$X[i][j]$均指变化后的。

给每一个课程建立$M-1$个点$a_{i,1}, a_{i,2}, \ldots, a_{i,M-1}$，并令$a_{i,0} = S, a_{i,M} = T$。$a_{i,j-1}$向$a_{i,j}$之间连一条容量为$X[i][j]$的边($1 \le j \le M$)。

很容易看出此时的最小割就是没有K个条件时的最优解，割掉一条边$a_{i,j-1} \to a_{i,j}$意味着在第j个学期选择课程i。考虑一个条件(A_i, B_i)和一个学期t，我们连一条从$a_{A_i,t-1}$到$a_{B_i,t}$的边，容量为$+\infty$，代表“A_i在学期t前选修”和“B_i在学期t后选修”中至少有一个成立。

之后跑最小割就是答案。

时间复杂度：$O(\text{networkflow}(NM, NM + KM))$

空间复杂度：$O(NM + KM)$

§20 BWGAME

§20.1 题目描述

一个$N \times N$的矩阵我们用两个长度为N的数组L, R表示：$A_{i,j} = [L_i \le j \le R_i](1 \le i, j \le N)$。求此矩阵的行列式的符号(正，负，零)。

§20.2 数据范围

数据组数不超过15，$1 \le N \le 10^5, 1 \le L_i \le R_i \le N$。

§20.3 解题报告

考虑高斯消元，我们希望在消元的的过程中每一行仍然保持性质“连续的一段是1，其他都是0”。我们用L_i, R_i来描述一行，每一行以R_i为关键字存入编号为L_i的集合。消元到变量i时：

- 选出集合i的最小元素R_i。如果$R_i < i$那么丢掉这个元素继续选，直到$R_i \ge i$或集合为空。如果集合为空，那么返回0。
- 把这个集合与编号为$R_i + 1$的集合合并。这代表两行相减的过程。
- 更新行列式的符号。

用可并堆维护集合。我用的是配对堆，效果不错。
时间复杂度：$O(TN \log N)$
空间复杂度：$O(N)$

§21 ANUDTQ

§21.1 题目描述

给定一棵初始时有N个点的有根树，每个点有点权。支持四种操作共M个：

- $1\ x\ y$：插入一个权值为y的节点作为节点x的孩子；
- $2\ x\ y$：给x的子树内所有节点的点权加y；
- $3\ x$：删除子树x；
- $4\ x$：求子树x内所有节点的权值和。

强制在线。

§21.2 数据范围

$1 \le N, M \le 10^5$，一个节点的权值在$[-1000, 1000]$内。

§21.3 解题报告

由于所有询问都涉及子树，所以可以考虑维护树的dfs序。我们定义dfs序为：在dfs某个节点x的时候，进入x和离开x时各向dfs序添加一个x，添加的两个节点分别称作l_x和r_x。区间$[l_x, r_x]$构成了x的子树。

$1\ x\ y$：设新插入的节点编号为N，直接在l_x后面插入l_N和r_N，其权值均为y；
$2\ x\ y$：给区间$[l_x, r_x]$的权值增加y，直接区间加法；
$3\ x$：删除区间$[l_x, r_x]$；
$4\ x$：求区间$[l_x, r_x]$的和。
用一个普通的splay来维护dfs序就可以了。
时间复杂度：$O((N + M) \log N)$
空间复杂度：$O(N + M)$

§22 CNTDSETS

§22.1 题目描述

N维空间中的点$(x_1, x_2, \ldots, x_N)$与$(y_1, y_2, \ldots, y_N)$的Chebyshev距离被定义为

$$\max_{i=1}^{n}(|x_i - y_i|)$$

点集的直径被定义为点集中任意两点的Chebyshev距离的最大值。

求N维空间中直径为D的点集的个数，**平移后相同的算一种**。模10^9+7。

§22.2 数据范围

数据组数不超过10，$1 \le N \le 1000$，$1 \le D \le 10^9$。

§22.3 解题报告

因为平移后相同的算一种，所以我们可以取一种集合的代表元。我取的是各维坐标的最小值都恰为0的点集。

即，求满足以下条件的N维点集个数：

- 1.所有点的所有坐标都在$[0, D]$内；
- 2.对于每一个$1 \le i \le N$，存在一个点其第i维坐标为0；
- 3.存在一个点的某个坐标为D。

首先考虑第3条，我们用总体情况减去"所有点的所有坐标都在$[0, D-1]$内"的情况即可。故下面的讨论中我们无视第3条。

我们发现第2条比较难以实现。使用容斥，枚举哪些维度不满足条件。事实上我们只需要知道不满足条件的维度个数，设为i，那么答案就加上

$$(-1)^i \binom{n}{i} 2^{D^i (D+1)^{N-i}}$$

通过$O(N)$预处理D^i和$(D+1)^i$模$p-1$($p = 10^9+7$)的值，运用费马小定理，这个式子是可以在$O(\log p)$的情况下求出来的。而模大质数的组合数是可以$O(N)$预处理，$O(1)$询问的。

枚举i，可以在$O(N \log p)$的时间内求出答案。

本题的时间复杂度为$O(TN \log p)$，不知道为什么出题人只把N放到了10^3。

时间复杂度：$O(TN \log p)$

空间复杂度：$O(N)$

§23 TREECNT2

§23.1 题目描述

给出一棵N个点的带边权无根树，求这样的点对$(x, y)(x \ne y)$的个数，x到y的路径上的边的最大公约数为1。

需要支持Q次对边权的修改，每次修改后给出答案。

§23.2 数据范围

$1 \le N \le 10^5, 1 \le Q \le 100$，边权不超过$10^6$。

§23.3 解题报告

通过莫比乌斯反演，我们可以得到答案就是，对于每一个$t \in \mathbb{Z}_+$，统计"每条边权值都是t的倍数"的路径条数，乘上$\mu(t)$，求和。

对于每一个数t，我们维护一个并查集$DSU(t)$表示只考虑权值是t的倍数的边时，这N个点的连通情况。对于一个并查集，如果大小为i的连通块有P_i个，那么答案为$\sum_{i=1}^{N} P_i \frac{i(i-1)}{2}$。如果使用动态开节点的并查集那么空间复杂度可以承受。

现在考虑修改。首先我们可以让并查集只考虑所有从未被修改过的边。计算询问的时候遍历所有修改过的边(不超过Q条)，设此时其权值为w，再找一个$t|w$，让$DSU(t)$中这条边相连，连边的时候更新答案。最后将并查集复原即可，使用不带路径压缩的并查集可以做到复原。

设A是权值的大小，$\tau(O(A))$表示小于等于A的数的因子个数的数量级。

时间复杂度：$O(A\log A+\tau(O(A))(N\alpha(N)+Q^2\log N))$
空间复杂度：$O(A\log A+\tau(O(A))(N+Q))$

§24 KNGHTMOV

§24.1 题目描述

有一个骑士在一个无穷大的棋盘上行走。起初它在$(0,0)$，现在它要求移动到(X,Y)。它只允许走向量(A_x,A_y)或(B_x,B_y)。另外，有K个点是不能被经过的。求行走的方案数，如果是$+\infty$那么输出-1。

§24.2 数据范围

数据组数不超过5，$0\le K\le 15$，其他数在$[-500,500]$之间。$(0,0),(X,Y)$不在那K个点中。

§24.3 解题报告

首先考虑A,B不共线的情况。此时不会走出循环，即答案不可能是$+\infty$。

考虑一个目的地$P=(X,Y)$，没有"不准到某个点"的限制，求走到P的方案数，记为$ways(P)$。我们设走了C_A次向量A、C_B次向量B。

$$\begin{cases} X = & C_AA_x+C_BB_x \\ Y = & C_AA_y+C_BB_y \end{cases}$$

解出C_A,C_B之后，如果它们都是非负整数，那么答案显然是$ways(P)=\binom{C_A+C_B}{C_A}$。

考虑添加了"不准到一些点"的限制之后，可以用$O(K^2)$的容斥原理dp求出方案数。具体地，我们把目标点和被禁止的点称为关键点，第i个关键点记为P_i。用f_i表示从$(0,0)$走到P_i，中途不经过其他关键点的方案数。首先f_i赋值为$ways(P_i)$，然后考虑经过了关键点的情况，枚举经过的第一个点P_j，那么方案数为$f_jways(P_i-P_j)$，减掉就好。

这样就可以用$O(K^2)$的时间解决A,B不共线的情况。

讨论A,B共线的情况。有可能出现以下几种情形：

- A,B都是零向量，当且仅当$X=Y=0$时答案为$+\infty$，否则答案为0；
- (X,Y)与A或B不共线。答案显然是0。

其他情况都可以转化到一个数轴上面去，且所有数的坐标在$[-500,500]$内。现在问题变成了，给出一根无穷长的数轴、两种行走方式$+A,+B$，以及一些不可经过的点，问从0出发是否能到某个点。

我们考虑把数轴上$[-1001,1001]$内的所有点抽出来形成一个图，每个点x与$x+A,x+B$连有向边；如果$AB\le 0$，把±1001向自己连边，表示往外乱走可以走出自环。保留所有**能走到目标，且能从0走到**的点，然后做拓扑排序，如果发现环那么答案就是$+\infty$；否则我们可以直接在拓扑图上dp，得到方案数。

本题有许多trick，稍微不注意就容易跪在上面。比如，注意$A=B$的特殊情况；注意组合数开到$10^6=(2\times 500)^2$。

时间复杂度：$O(K^2)$或$O(A)$($A=1000$是坐标范围的大小)
空间复杂度：$O(K+A)$

§25 CBAL

§25.1 题目描述

一个字符串被称为平衡的，当且仅当a∼z中的每一个字符在其中都出现了偶数次。

给出一个母串T，$N=|T|$。

一组询问$[L,R,type]$的答案为：如果一个子串$T_{i,j}$是平衡的，且$L\le i<j\le R$，那么它对答案的贡献是$(j-i+1)^{type}$；否则它不影响答案。

给出Q组询问，依次输出它们的答案。要求在线算法。

§25.2 数据范围

数据组数不超过10^5，串长总和与询问总数都不超过10^5。$type \in \{0,1,2\}$。

§25.3 解题报告

对于每一个位置$0 \le i \le N$，我们定义它的特征值S_i为一个26位的01向量，表示$T_1, T_2, \ldots, T_i$中，每个字母出现次数的奇偶性。一个子串$T_{l,r}$是平衡的，当且仅当$S_{l-1} = S_r$。

询问变成了：考虑$L-1 \le i < j \le R$且$S_i = S_j$的所有(i,j)，求$(j-i)^{type}$的和。

因为我们只需要判断S的两个元素是否相等，所以可以将S离散化，使得其中元素都在$0 \sim N$内。

我们考虑$O(\sqrt{n})$分块，记录这么一些信息：

- $f_{i,j,type}$：对于一组询问，L为块i的开头，R为块j的结尾，答案是多少；
- $cnt_{b,s,type}$：前b个块的所有元素i中，$\sum_{S_i=s} i^{type}$。

首先我们考虑在知道一个询问$[L,R,type]$的答案ans，且对于任意的$(s,type)$知道

$$CNT_{s,type} = \sum_{L-1 \le i \le R} [S_i = s] i^{type}$$

的情况下，如何求出$[L,R+1,type]$的答案。事实上我们只需要访问$CNT_{S_{R+1}}$即可，是$O(1)$的。我们同时也可以更新$CNT_{S_{R+1}}$，也是$O(1)$的。

预处理f：枚举i，从小到大枚举j，用上述$O(1)$的方法维护答案和CNT数组。

回答询问：首先考虑完全被覆盖的块，读出f值得到答案；然后从左边和右边添加剩余的节点。

时间复杂度：$O((N+Q)\sqrt{N})$

空间复杂度：$O(N\sqrt{N})$

§26 QTREE

§26.1 题目描述

给出一个N个点N条边的无向连通图。这个图显然只有一个环，我们假设环上的点的数目是奇数。我们定义两点之间的最短路为经过边数最少的路径，这样任意两点之间的最短路显然是唯一的。每条边同时还有边权。

要求支持两种操作共Q个：

- f u v：将u到v最短路上的边的权值取反。
- ? u v：求u到v最短路上的边权的最大连续子段和。

§26.2 数据范围

$N, Q \le 10^5$，边权的绝对值不超过10^4。

§26.3 解题报告

考虑树上的版本，树链剖分套线段树，线段树中每个节点记录其

- 和
- 最大/最小前缀和
- 最大/最小后缀和
- 最大/最小子段和

可以很容易地合并值。由于同时记录了最小的值，所以打标记的时候可以快速更新。

考虑有环的情况。把环边e挑出来单独处理，剩下就是一棵树的情况，可以直接做。

修改：直接在树上搞；如果待修改路径经过e那么直接修改e。

询问：如果没有经过e那么直接搞；否则搞出e左右两边的信息与e合并。

时间复杂度：$O(N + Q\log^2 N)$

空间复杂度：$O(N)$

§27 challenge GERALD09

§27.1 题目描述

给定一个$N \times M$的空矩阵，填上G,C,A,T以构成一个互不相同的子矩阵数目尽可能接近K的基因矩阵。

§27.2 数据范围

数据组数不超过100，$N, M \le 15$。K是$[1, N^2M^2]$内随机生成的。

§27.3 解题报告

考虑给出一个矩阵A，如何求出其不相同的子矩阵个数$\text{ans}(A)$。直接用矩阵哈希可以做到$O(N^2M^2)$的复杂度。

我们可以使用一种贪心方法生成一个$\text{ans}(A)$很接近K的矩阵。一开始A是全A矩阵，逐位随机产生$A_{i,j}$的值，如果$\text{ans}(A) > K$则把$A_{i,j}$赋为0。这个做法复杂度是$O(N^3M^3)$的，无法承受。但是可以用二分的思想，每次把剩下的0中的一半生成，如果$\text{ans}(A)$超过了K，那么就把这一半赋为0。

但是这样不一定是最好的。我们可以考虑另外一种方法：每次确定一个元素$A_{i,j}$，只考虑**所有元素都已经被确定的矩阵**中，不同的矩阵个数。确定一个元素后可以$O(NM)$地算出贡献。枚举当前这个元素是ATCG的哪一位，估算出应该有的不同矩阵个数$k_{i,j}$，取最接近$k_{i,j}$的作为这一位。

一个较好的$k_{i,j}$为(前提是：如果(i,j)已经被确定了，(k,m)没有被确定，那么$\neg(k \le i \wedge m \le j)$)

$$k_{i,j} = \frac{K\sum_{i\le n, j\le m} ij}{\sum_{(i,j)\text{已被确定}} ij}$$

当然还有其他的求出初始解A的方法。不同的方法精确度不同，效率也不同。

由于生成的初始解一般不是很优，我们还要考虑随机调整我们的解。每次随机改变A中一个元素，如果能够导致更优解产生，那么直接接受该解；否则不接受(即爬山)或者以一定概率接受(退火?)。调整的次数也有讲究，一般来说与TN^2M^2成反比，有时候可以人为改变这个参数(实际运行时间较小的时候，或者K比较小的时候，调整次数变大)。

这样就可以取得不错的分数了。

时间复杂度：$O(N^2M^2(\log N + \log M) + \mathit{count} \times N^2M^2)$

空间复杂度：$O(N^2M^2)$

§28 BAKE

§28.1 题目描述

有10种不同的商品，每种有3种尺寸。这些商品将被销售到10个省份去。每个省份有不超过20座城市，每座城市有不超过5个地区。支持两种操作共S个：

- I product_id[.size_id] province_id[.city_id[.region_id]] M/F age units_sold 表示往一个地方的指定顾客(给出了TA的性别和年龄)销售了units_sold个某种类型的产品。方框内的参数可能缺失。units_sold$\le$100。

- Q product_id[.size_id] province_id[.city_id[.region_id]] M/F start_age[-end_age] 查询满足指定条件的销售记录的产品数之和。当[-end_age]缺失时表示顾客年龄等于start_age。product_id为-1时表示查询所有产品；province_id为-1时表示查询所有地区。

§28.2 数据范围

$S \leq 10^5$。

§28.3 解题报告

直接模拟。对每一个产品、地区、性别、年龄或它们的组合，记录其销售总量。
时间复杂度：$O(S)$
空间复杂度：$O(10 \times 3 \times 10 \times 20 \times 5 \times 2 \times 90 \times \text{sizeof}(\textbf{int}))$

§29 ANDOOR

§29.1 题目描述

给出一个矩形，矩形内的颜色都是白色，其他都是黑色。把矩形中N个圆染成黑色，求白色部分的圆弧周长(即，不算矩形框)。

§29.2 数据范围

数据组数不超过1000，$N \leq 1000$，各组数据的N之和不超过5000。坐标范围$[0, 1000]$，半径不超过1000。

§29.3 解题报告

对于每一个圆，我们枚举其他的圆(和矩形框)可以得到其哪些区间被覆盖了。接下来做一遍线段覆盖即可得到它没有被覆盖的圆弧的总长。加起来即可。

这个题卡精度，在判断两个圆相切的时候我们不能用实数去比较。因为实数比较会产生大约10^{-17}的精度误差，而经过arccos函数后，这个误差会被放大到10^{-8}。用整数比较就可以AC 了。
时间复杂度：$O(\sum N^2 \log N)$
空间复杂度：$O(N)$

§30 PARADE

§30.1 题目描述

给定一张N个点M条边的有向图，每条边有一个$cost$。要求选出一些英雄进行游行。每个英雄的路线是图中的一条路径。费用是下列三项的和：

- 每条路经过的次数乘上它的$cost$；
- 对于起点不等于终点的每个英雄，支付C的费用；
- 对于没有被经过的点，支付C的费用。

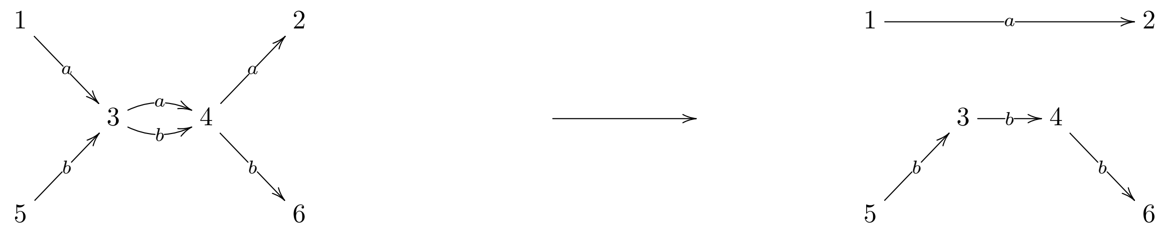
图和边权是固定的，有K组询问，每组询问给了C要求最小费用。

§30.2 数据范围

$2 \leq N \leq 250, 1 \leq M \leq 30000, 1 \leq K \leq 10000, 1 \leq cost, C \leq 10000$。

§30.3 解题报告

很神的一个思维题。



上左图中英雄a和英雄b都经过了$3,4$这两个点。我们考虑在1到2之间连一条长度为$cost(1,3)+cost(3,4)+cost(4,2)$的边，让英雄$a$直接从1走到2，如上右图。这样做的好处是，每个点至多被经过一次，且答案不变。

首先我们进行floyd，对于每两个点i,j添加一条边表示它们的最短路径$dist(i,j)$。然后我们规定一个点只能被经过至多一次。

现在问题变成了：求一个边集，满足每个点的入度、出度至多为1。考虑代价，第1部分就是选的边的$cost$之和；第2部分就是入度为0，出度不是0的点数乘C；第3部分就是入度、出度都是0的点数乘C。$2,3$部分加起来就是入度为0的点数乘C。

考虑费用流模型。对于图中每一个点i，我们将其拆成两个点$l(i),r(i)$。如果$i \neq j$，连一条费用为$dist(i,j)$的边$l(i) \to r(j)$。源点向所有$l(i)$连费用为0的边，所有$r(i)$向汇点连费用为0的边。所有边容量为1。设流量为i的时候，最小费用为f_i，那么答案就是所有的$f_i+(n-i)C$中的最小值。

我们预处理出所有f_i，询问的时候暴力搞就好。运用一些性质是可以做到$O(\log N)$一次询问的，不过我并没有写。

时间复杂度：$O(N^3+\text{costflow}(2N+2,N^2+2N)+KN)$

空间复杂度：$O(N^2)$

§31 MATCH

§31.1 题目描述

给出一个$N \times M$的完全二分图，以及每条边的出现概率。求这个二分图的期望最大匹配大小。

§31.2 数据范围

$N \leq 5, M \leq 100$。

§31.3 解题报告

考虑Hall定理。

定理1. 对于一张二分图$G=(X,Y,E)$，G存在一个大小为$|X|$的匹配，当且仅当对任意集合$W \subseteq X$，$N_G(W) \geq |W|$，其中$N_G(W)$表示Y中与W中至少一个点相邻的点的数量。

我们考虑令$f_{x,i}$表示考虑右边的点中编号$\leq i$的点，状态为x的概率。记c为状态数。

状态被定义如下：对于每一个集合$S \subseteq X$我们记录一个$0\ldots|S|$的整数$x(S)$表示$\min(|S|,N_G(S))$。

看起来状态很多，实际上合法状态只有$c=9171$个。我们把初始状态加入队列，每次从队列中拿出一个状态进行转移，这样转出来的都是合法状态。这样就可以进行转移了。

时间复杂度：$O(c2^N(M+2^N))$

空间复杂度：$O(Mc)$

§32 QRECT

§32.1 题目描述

维护一个平面，支持三种操作：

- 插入一个矩形；

- 删除一个之前插入的矩形；
- 询问一个给定矩形与多少个之前插入的矩形有公共点。

§32.2 数据范围

操作个数不超过10^5。

§32.3 解题报告

两个矩形(x_1,y_1,x_2,y_2)与(x_3,y_3,x_4,y_4)相交，当且仅当

$$[x_1,x_2]\cap[x_3,x_4]\neq\emptyset\wedge[y_1,y_2]\cap[y_3,y_4]\neq\emptyset$$

考虑一次询问，我们要求与给定矩形(X_1,Y_1,X_2,Y_2)有公共点的矩形个数。我们转化成求没有公共点的矩形个数，设所求矩形为(x_1,y_1,x_2,y_2)，那么可以看成以下8项的贡献：

(1) $x_2<X_1$的矩形个数
(2) $y_2<Y_1$的矩形个数
(3) $x_1>X_2$的矩形个数
(4) $y_1>Y_2$的矩形个数
(5) $x_2<X_1$且$y_2<Y_1$的矩形个数(贡献为负，因为被算了两次)
(6) $x_2<X_1$且$y_1>Y_2$的矩形个数(贡献为负，因为被算了两次)
(7) $x_1>X_2$且$y_2<Y_1$的矩形个数(贡献为负，因为被算了两次)
(8) $x_1>X_2$且$y_1>Y_2$的矩形个数(贡献为负，因为被算了两次)

维护四个“矩形数点问题”(即，插入/删除一个点，询问矩形内的点数)，分别代表(5), (6), (7), (8)。插入一个矩形(x_1,y_1,x_2,y_2)时，分别在其中插入$(x_2,y_2),(x_2,y_1),(x_1,y_2),(x_1,y_1)$。“矩形数点问题”是经典问题，可以用离线分治+扫描线在$O(\log^2 Q)$一次询问的时间内得到解决。

时间复杂度：$O(Q\log^2 Q)$

空间复杂度：$O(Q\log Q)$

§33 TRIPS

§33.1 题目描述

给出一棵N个点的树T，每条边的权值是1或2。有Q个学生，第i个学生要从s_i走到t_i，他的体力为p_i。每一天学生尽量走，如果走下一个节点会使得当天行程大于p_i那么就在当前节点休息到下一天。对每个学生输出他需要走多少天到目的地。

§33.2 数据范围

$N,Q\leq 10^5$。

§33.3 解题报告

考虑询问(s,t,p)，假设我们知道了s和t的LCA，记为w，那么可以这么解决问题：

- 算出s_1表示最大的数，满足从s出发走s_1天走的最大距离不超过s到w的距离；
- 算出s_2表示在上述情况下s_1天后学生与w的距离；
- 同理算出t_1,t_2；
- 答案等于$s_1+t_1+\lceil\frac{s_2+t_2}{p}\rceil$。

令$H=O(\sqrt{N})$。

假设$p>H$，则答案不超过$\frac{N}{H}$。直接暴力跳求出s_1,s_2,t_1,t_2，复杂度$O(\frac{N}{H}\log N)$。

假设$p \le H$，我们预处理出$f_{i,j,k}$表示从点i开始，$p = j$，走2^k天走到的点。这个复杂度是$O(NH\log N)$的。询问的时候直接倍增即可，复杂度$O(\log N)$。

如果离线，把询问按照p排序，那么可以将空间复杂度从$O(NH\log N)$优化到$O(N\log N)$。

时间复杂度：$O((N+Q)\sqrt{N}\log N)$

空间复杂度：$O(N\log N)$

§34 CLOWAY

§34.1 题目描述

对于两个图(都是无向简单图)$G_1 = (V_1, E_1), G_2 = (V_2, E_2)$，定义$G_1 \boxtimes G_2$是这样一个图$(V, E)$，其中$V = V_1 \times V_2$，如果$u, x \in V_1, v, y \in V_2$，那么$(u, v) - (x, y) \in E$当且仅当下列三项之一：

- $u = x \wedge v - y \in E_2$;
- $u - x \in E_1 \wedge v = y$;
- $u - x \in E_1 \wedge v - y \in E_2$

那么$\boxtimes$满足结合律。

有T个图$G_1, G_2, \ldots, G_T$。每次给出l, r, K，令G为$G_l \boxtimes G_{l+1} \boxtimes \cdots \boxtimes G_r$。求$G$中长度不超过$K$的回路个数。模$10^9+7$。这样的询问有$Q$个。

§34.2 数据范围

$T \le 50$，每个图的点数都不超过50，$Q \le 2 \times 10^5$，$1 \le l \le r \le T$，$1 \le K \le 10^4$。

§34.3 解题报告

令$w_{i,j}$为第i张图中长度为j的回路的个数。

考虑预处理w数组，可以暴力求出$w_{i,1} \ldots w_{i,n}$，而根据Cayley-Hamilton定理，设邻接矩阵的特征多项式为$\sum_{i=0}^{n} c_i x^i$，则$G^{j+n} = -\frac{\sum_{i=0}^{n-1} c_i G^{j+i}}{c_n}$。故可以在$O(N^4 + KN)$的时间内求出所有$w_i$。

特征多项式的话，代入$O(N)$个值求一下行列式，然后插值就好了。

假设询问1 N K，答案就是(容斥)

$$\sum_{k=0}^{K} (-1)^k \binom{K}{k} \prod_{i=1}^{N} \left(\sum_{j=0}^{K-k} \binom{K-k}{j} w_{i,j} \right)$$

令

$$r_{i,k} = \sum_{j=0}^{k} \binom{k}{j} w_{i,j}$$

可以用FFT预处理出每张图的r数组(下标到K即可)。

询问的时候将待询问图的r数组相乘，拆开组合数就可以FFT了。对于l, r相同的询问只需做一遍FFT，然后前缀和，对每个询问找答案即可。

再就是模数的问题了，我是取了一个10^9级别的数和一个10^{16}级别的数做NTT模数，然后用中国剩余定理合并答案。

时间复杂度：$O(TN^4 + (TN + T^2)K\log K + Q)$

空间复杂度：$O(NK + TN^2)$

§35 EASYEX

§35.1 题目描述

有一个K面骰子，上面分别写着$1 \ldots K$。还有两个整数$L, F(0 < L \le K)$。掷N次骰子，记掷出数字i的次数为a_i。求$\prod_{i=1}^{L} a_i^F$的期望值模2003。

§35.2 数据范围

数据组数不超过50，$0 < N, K \le 10^9, 0 < LF \le 50000, 0 < F \le 1000$。

§35.3 解题报告

记$\mathrm{MAT}(i,j,k)$为i行j列每个元素都在$[1,k]$之间的矩阵的集合。答案等于

$$\begin{aligned} Ans &= \frac{1}{K^N}\sum_{r\in \mathrm{MAT}(1,N,K)}\prod_{i=1}^{L}(\sum_{j=1}^{N}[r_j=i])^F \\ &= \frac{1}{K^N}\sum_{r\in \mathrm{MAT}(1,N,K)}\sum_{j\in \mathrm{MAT}(L,F,N)}\prod_{i=1}^{L}\prod_{p=1}^{F}[r_{j_{i,p}}=i] \end{aligned}$$

对所有矩阵j计算满足条件的向量r的个数：对于一个矩阵j而言，如果存在一个元素在j的不同行中出现了，那么其答案就是0；否则答案就是$K^{N-cnt(j)}$，其中$cnt(j)$为j中不同元素的个数。

我们令c_i为j中第i行的不同数的个数。给出c，我们可以得到j对答案的贡献和满足条件的j的个数。

可以得到$cnt(j)=\sum_{i=1}^{L}c_i$。j对答案的贡献就是$K^{N-cnt(j)}$。

根据容斥原理，第x行的方案数为

$$f(c_x,F)=\sum_{i=0}^{c_x}\binom{c_x}{i}(-1)^{c_x-i}i^F$$

故j的个数就是

$$\frac{N!}{(N-cnt(j))!}\prod_{i=1}^{L}\frac{f(c_i,F)}{c_i!}$$

现在答案等于，对于所有的c，下面式子的和(记$Cnt(c)$为c中元素的和)

$$\frac{1}{K^{Cnt(c)}}\frac{N!}{(N-Cnt(c))!}\prod_{i=1}^{L}\frac{f(c_i,F)}{c_i!}$$

我们用$dp_{i,j}$表示长度为i，Cnt值为j的序列，$\prod_{t=1}^{i}\frac{f(c_t,F)}{c_t!}$的和。

$$dp_{i,j}=\sum_{k=1}^{\min(j,F)}dp_{i-1,j-k}\frac{f(k,F)}{k!}$$

可以写成$\{\frac{f(k,F)}{k!}\}$序列的L次方(乘法为卷积)。

令$M=2003$为模数，我们发现$\frac{N!}{(N-cnt(j))!}$当$cnt(j)\ge M$的时候等于0。故我们枚举$cnt(j)$时可以只考虑$cnt(j)<M$的情况。即，对于$dp_{i,j}$，我们可以认为$j<M$。

f值可以$O(F^2)$暴力求(也许可以用FFT吧，不过这里不是复杂度瓶颈)。dp数组可以用$O(M\log M\log L)$的FFT+快速幂。

时间复杂度：$O(T(M\log M\log L+F^2))$

空间复杂度：$O(LF)$

§36 RNG

§36.1 题目描述

给出$A_1,A_2,\dots,A_k,C_1,C_2,\dots,C_k,N$，令$i>k$时$A_i=\sum_{j=1}^{k}A_{i-j}C_j$。求$A_N \bmod 104857601$。

§36.2 数据范围

$1\le N\le 10^{18}, k\le 300000$。

§36.3 解题报告

考虑复杂度$O(k^3\log n)$的矩阵快速幂算法。令

$$\mathbf{M}=\begin{bmatrix}C_1 & C_2 & C_3 & \cdots & C_{k-1} & C_k\\ 1 & 0 & 0 & \cdots & 0 & 0\\ 0 & 1 & 0 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & 1 & 0\end{bmatrix}$$

$$\mathbf{X}_i=\begin{bmatrix}A_i & A_{i-1} & \cdots & A_{i-k+1}\end{bmatrix}$$

那么$\mathbf{X}_i=\mathbf{X}_{i-1}\mathbf{M}$。故$\mathbf{X}_n=\mathbf{X}_k\mathbf{M}^{n-k}$。可以使用$O(k^3\log n)$的矩阵快速幂解决问题。

但是这样显然太慢。根据Cayley-Hamilton定理，

$$p(\mathbf{M})=\mathbf{M}^k-\sum_{i=0}^{k-1}C_{i+1}\mathbf{M}^i=\mathbf{0}$$

可以得到一个推论，即$\mathbf{M}^i(i\in\mathbb{Z}_+)$可以表示成$\mathbf{M}^0,\mathbf{M}^1,\ldots,\mathbf{M}^{k-1}$的线性表示。不妨设

$$\mathbf{M}^i=\sum_{j=0}^{k-1}f_{i,j}\mathbf{M}^j$$

如果我们能在知道f_i的情况下推出f_{2i}和f_{2i+1}，那么通过快速幂我们可以求出答案。而

$$\mathbf{M}^{2i}=(\mathbf{M}^i)^2=\sum_{j_1=0}^{k-1}\sum_{j_2=0}^{k-1}f_{i,j_1}f_{i,j_2}\mathbf{M}^{j_1+j_2}$$

即，一次多项式乘法可以将$\mathbf{M}^{2i}$($\mathbf{M}^{2i+1}$类似)表示成$\mathbf{M}^0,\mathbf{M}^1,\ldots,\mathbf{M}^{2k-2}$的线性表示。现在关键是把$\mathbf{M}^i(i\geq k)$的线性表示代入，即可得到$\mathbf{M}^0,\mathbf{M}^1,\ldots,\mathbf{M}^{k-1}$的线性表示。我们将这个多项式除以特征多项式得到余数，就是我们要求的f_{2i}。

多项式除法可以做到复杂度$O(k\log k)$。

假设我们已经知道了f_{n-k}，要计算答案，可以证明答案就是$\sum_{j=0}^{k-1}f_{n-k,j}A_{j+k}$。将这个多项式除以特征多项式得到余数$f'$，答案就等于$\sum_{j=0}^{k-1}f'_{n-k,j}A_j$。

时间复杂度：$O(k\log k\log n)$

空间复杂度：$O(k)$

§37 HAMILG

§37.1 题目描述

Askar和Bob用一个棋子在一个无向图G上玩一个游戏。游戏是这样的：

- Askar选定一个节点作为起点，将棋子放在这个点上
- 然后双方开始轮流行动，Bob先行
- 轮到行动的一方需要将棋子沿着某一条边移动到另一个节点上
- 棋子不能移动到一个之前放过棋子的点，包括起点
- 无法行动的一方就输了

如果Askar选定节点v作为起点之后能够获胜，我们就称点v为必胜点。我们假设双方都按最优策略行动。给定图G，请你找出图中必胜点的数量。

§37.2 数据范围

数据组数不超过100，边数之和不超过10^6，$1\leq N\leq 2000$，图是连通图。

§37.3 解题报告

假设Askar的第一步已经确定了。

一方面，如果存在一个最大匹配M使得M不包含棋子的初始位置，那么Askar有必胜策略。因为Bob的每一步之后，棋子都会在M的某个匹配点x上，且与x匹配的点y未被经过——否则就相当于找到了一条增广路，M不是最大匹配了。而Askar的策略就是走向y。

另一方面，如果棋子的初始位置p一定在所有最大匹配中，那么Bob有必胜策略。考虑一个最大匹配M，Bob每次沿着M的匹配边来移动棋子。Askar无论怎么走下一步都一定会走到一个匹配点——否则就相当于找到了一条长度为偶数、匹配边和非匹配边交错的一条路L，且这条路的起点是p。这样$M \oplus L$是一个不包含p的最大匹配。

故对于给定的图我们只需要求出不一定在最大匹配中的点数。考虑求出一个最大匹配，然后将每一个未盖点放入队列来寻找增广路，如果对于一个点x，存在一个未盖点y，使得以y为起点寻找增广路时，x是被遍历过的偶点，那么x是一个必胜点。

时间复杂度：$O(TN^3)$，不过带花树的常数太小了，跑得过。

空间复杂度：$O(M)$

§38 GTHRONES

§38.1 题目描述

两个人玩一个游戏：纸上写着一些数，一个数可能出现多次。有N种不同的数，第i种数为u_i，出现c_i次。

第一局，先手选定一个数，这个数被称作“current number”(简称CN)。

从第二局(第二局由后手执行)开始，行动方在纸上擦掉CN，如果有多个那么只擦掉一个；然后重新选一个数为CN。如果这一局之前的CN为u，重新选为CN的数为v，那么一个合法的操作必须满足：

$$v > u, u|v\text{且}\frac{v}{u}\text{是质数；或}u > v, v|u\text{且}\frac{u}{v}\text{是质数。}$$

(1不是质数)

不能移动的人输。

问谁赢。如果先手赢，求出他第一步应该选哪个数为CN，若有多个求出最小的那个。

§38.2 数据范围

$1 \le N \le 500, 1 \le u_i \le 10^{18}, 1 \le c_i \le 10^9$。

§38.3 解题报告

见**HAMILG**一题解题报告。

判断两个点是否满足条件，这个可以用Miller-Rabin算法完成，设值域为A那么可以使用$O(N^2)$次Miller-Rabin判断两两点对是否满足条件，复杂度$O(N^2 \log A)$。

把满足条件的u, v相连，发现这个图是二分图，求出哪些点可以不在最大匹配中即可。

数的个数可能很多，需要缩点后跑网络流，求出一个最大匹配。

求关键点(必须在匹配中的点)的算法如下：考虑左边的关键点，以右边的所有未盖点为起点进行dfs，只允许沿着非匹配边从右走到左，沿着匹配边从左走到右(相当于给边定向)。dfs未经过的所有左边点都是关键点。因为dfs的过程相当于找增广路，这个算法是正确的。右边的关键点类似。

时间复杂度：$O(\text{networkflow}(N, N^2) + N^2 \log A)$

空间复杂度：$O(N^2)$

§39 CONPOIN

§39.1 题目描述

问一个图是不是极大平面图。即，该图是一个平面图，且其任意一个平面嵌入都满足再加入一条连接两点的直线段之后该图不是平面图。

§39.2 数据范围

数据组数不超过10，$N \le 7\times 10^4, M \le 3\times 10^5$。

§39.3 解题报告

论文题。论文见

`http://zh.scribd.com/doc/269955060/A-simple-recognition-of-maximal-planar-graphs`。

对于一个平面图我们认为$M = O(N)$。事实上，对于$N \ge 3$的极大平面图，$M = 3N - 6$。

构造出一个点的排列v，使得$v_1, v_2, \ldots, v_i (i \ge 3)$组成一个双连通平面图$G_i$，$v_{i+1}$在$G_i$的外部，且$v_{i+1}$向$G_i$连的边是$G_i$的外边界的一段。我们称$v$是$G$的一个CO(canonical ordering)。

用函数ORDER构造出G的一个CO，从v_n到v_3构造。令v_i为当前点，VC为$v_1, v_2, \ldots, v_i$的外边界。每次找出外边界上一个点并将其删掉，维持G的每个面都是三角形的性质，更新VC。可以用C++的set维护$|N_{G'}(v) \cap VC| = 2$的所有点，复杂度$O(N\log N)$。

用函数EMBED给出图的一个平面嵌入。用有序数列VC表示G_i的外边界。如果v_i在G_{i-1}中的邻居在VC中不是连续的一段，那么v不是合法的CO；否则更新VC。如果用splay维护VC数组的话可以做到$O(N\log N)$。(事实上暴力可以水过数据)

找到最小度数的点v_1，它的度数不超过5，固定一个邻居做v_2，枚举另一个邻居做v_n，调用ORDER和EMBED即可。

时间复杂度：$O(N\log N)$

空间复杂度：$O(N)$

§40 DIVIDEN

§40.1 题目描述

给出平面上一个N度的角，其中N是整数。使用尺规作图将其N等分。

§40.2 数据范围

坐标绝对值不超过10^3，$0 < N < 360$，输出的步数不能超过1000。

§40.3 解题报告

定理2. *有解的充要条件是$N \bmod 3 \ne 0$。*

Proof. 首先我们构造出3°。用尺规作正五边形的方法可以得到108°，即可得到18°；然后作出30°并等分，即可得到15°。两个角度相减得到3°。

然后我们证明无法得到1°。事实上能够得到$\frac{360}{N}^\circ (N \in \mathbb{Z})$的充要条件是，$N$可以表示成2的幂乘以一些不同的费马数的乘积。而360并不满足此条件。

对于$N \bmod 3 \ne 0$的情况，通过作出一系列3°角与N°角相减可以得到1°，即N等分了该角。

对于$N \bmod 3 = 0$的情况，假设能够N等分该角，那么我们可以作出一系列3°角，拼成N°角，然后将其N等分来得到1°。而这与上面的结论是矛盾的。 □

上述证明也给出了本题的构造思路。我们先作出3°角，然后依次作角得到3的倍数的Answer并得到1°角，再得出所有答案。操作次数应该是$O(N)$的。

时间复杂度：$O(N)$

空间复杂度：$O(N)$

§41 STREETTA

§41.1 题目描述

维护一个长度为N的序列a，序列的每一个元素是一个二元组$a_i = \{S_i, x_i\}$，其中S_i是一个数集，x_i是一个整数。最初每个二元组都是$\{\emptyset, 0\}$。支持三种操作：

- $1\ u\ v\ a\ b$：对于$0 \le i \le v-u$，$S_{u+i} \to S_{u+i} \cup \{b+ia\}$。
- $2\ u\ v\ a\ b$：对于$0 \le i \le v-u$，$x_{u+i} \to x_{u+i} + b + ia$。
- $3\ i$：求$\max(S_i) + x_i$。若$S_i = \emptyset$，输出`NA`。

§41.2 数据范围

$1 \le N \le 10^9, 1 \le M \le 3 \times 10^5$。

§41.3 解题报告

首先考虑维护x值。这样的标记支持加法，所以可以使用动态开节点的线段树维护。

维护$\max(S)$则稍微难一点。对线段树的每个节点$[l,r]$维护两个值a,b，表示这个节点与$ai+b$取了max。假设给这个节点打上一个新的标记$ci+d$。如果两个标记有一个总是大于另一个，那么直接替换；否则有四种情况：

- 在左子树中总有$ai+b \ge ci+d$；
- 在右子树中总有$ai+b \ge ci+d$；
- 在左子树中总有$ci+d \ge ai+b$；
- 在右子树中总有$ci+d \ge ai+b$。

我们以其中一种为例：在右子树中总是有$ci+d \ge ai+b$。那么我们把$ci+d$这个标记打在线段树上，用$ai+b$来更新左子树。即，我们总是把"大"得更多的那个标记打在整个树上，用另一个标记去更新其占优势的子树。

可以看出更新复杂度是$O(\log^2 N)$的。

时间复杂度：$O(M \log^2 N)$

空间复杂度：$O(M \log^2 N)$。

§42 challenge CLOSEST

§42.1 题目描述

三维空间中有N个给定点。有Q个询问点，你需要对每个询问点求出离它最近的给定点是谁。

你的分数与答对的询问数成正比。

§42.2 数据范围

$N = Q = 50000$，部分数据为随机生成，剩下的数据用来卡掉某些特定的算法。

§42.3 解题报告

我们使用一种类似于k-d树的思路。

给出一个点集S，我们按照某种方法将S分成两个点集S_l, S_r，并递归下去。对于一个点集S我们维护一些信息$h(S)$，使得在知道$h(S)$和询问点的坐标Q之后，我们可以尽快、尽量准确地求出Q到S中最近点的距离$d(Q,S)$的**估价**$d^*(Q,S)$。

询问一个点Q到点集S的最近距离时，实时维护当前得到的最小值$best$，如果$d^*(Q,S) \ge best$那么跳过S。否则，如果$d^*(Q,S_l) < d^*(Q,S_r)$，那么先查询$d(Q,S_l)$再查询$d(Q,S_r)$；否则先查询$d(Q,S_r)$再查询$d(Q,S_l)$。

如果$d^*(Q,S) \le d(Q,S)$，那么返回的答案总是正确的。由于本题中的答案只要不正确就没有意义，所以最好取$d^*(Q,S) \le d(Q,S)$。

我使用的是k-d树的写法，即处理出S的包围盒，以Q到包围盒的距离为$d^*(Q,S)$。这种做法对于$N=Q=50000$的随机数据可以在时限内得到所有的最优解。但是我把它卡掉了。具体地说，询问点是坐标范围$\pm 10^4$的随机点，给定点是距离原点10^7的随机点(即一个球面)，可以把我写的k-d树卡成暴力。

最高分(11月2日)使用的是另一种方法，对于一个点集取其中两随机点的中垂面将S分为两个部分，$d^*(Q,S)$可以取Q到中垂面的距离。这种做法虽然在随机数据不如k-d树，但是在球状数据和链状数据上的表现都非常好。

时间复杂度：$O(QN^{\frac{2}{3}})$

空间复杂度：$O(N)$

§43 RANKA

§43.1 题目描述

给出围棋的规则，要求在9×9的棋盘上构造一个长度为N的合法操作序列。

其中有一个要求就是，不允许两次出现相同的局面(防止全局同形再现)。

§43.2 数据范围

$N \le 10000$。

§43.3 解题报告

`http://senseis.xmp.net/?LongestPossibleGame`的神构造。

考虑让执黑子的一方填满棋盘中除了$(9,9)$以外的所有点。然后白子下在$(9,9)$，即可清空黑子。

接下来让黑子填满棋盘中除了$(9,9),(9,8)$以外的所有点，然后白子$(9,8)$，又可以清空黑子。

然后$(9,7),(9,6),\ldots$，直到白子填满了棋盘的后8行。这样可以轻松做到$N \le 5000$。

接下来交换黑白子的身份。黑子先清空白子并占领第1行，然后白子以

$$(9,1)\to(9,2)\to\cdots\to(9,9)\to(8,1)\to(8,2)\to\cdots\to(8,9)\to\cdots\to(2,9)$$

的顺序下棋(之所以第二维坐标要换一个顺序，是为了防止全局同形再现)，黑子再用一步清理掉所有白子，这样下去即可轻松做到$N \le 10000$。其中为了防止全局同形再现，有一些细节需要处理。

时间复杂度：$O(N)$

空间复杂度：$O(1)$

§44 SEAARC

§44.1 题目描述

给一个长为N的数组A，求$1\le i<j<k<l\le N, A_i=A_k\ne A_j=A_l$的$(i,j,k,l)$个数模$10^9+7$。

§44.2 数据范围

$1\le N\le 10^5, 1\le A_i\le 10^5$。

§44.3 解题报告

不考虑$A_i \ne A_j$的限制。

考虑出现次数较小($\le S$)的颜色，把这些颜色称作**小颜色**，其他颜色称作**大颜色**。

考虑$p=A_i$是大颜色的情况。考虑枚举$q=A_j$，设序列被颜色为p的元素分成了n段，前i段中颜色q的个数是$c_{q,i}$。我们要求的答案就是

$$\begin{aligned}answer &= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n-1}(c_{q,j}-c_{q,i})(c_{q,n}-c_{q,j}) \\ &= \sum_{i=1}^{n-1}\left(c_{q,n}c_{q,i}(2i-n)-c_{q,i}^2(i-\frac{1}{2})c_{q,i}^2\right)+\frac{1}{2}\left(\sum_{i=1}^{n-1}c_i\right)^2\end{aligned}$$

上面那个式子展示了对于一个颜色q我们需要关心哪些量，对于一个颜色p我们用$O(N)$求出所有颜色的这个量就可以算答案了。由于大颜色共有$O(\frac{N}{S})$个，所以复杂度为$O(\frac{N^2}{S})$。

考虑$q=A_j$是大颜色，但$p=A_i$是小颜色的情况。同上我们设序列被颜色为q的元素分成了n段，前i段中颜色p的个数是$c_{p,i}$。我们要求的答案是

$$\begin{aligned}answer &= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n-1}c_{p,i}(c_{p,j}-c_{p,i}) \\ &= \frac{1}{2}\left(\sum_{i=1}^{n-1}c_{p,i}\right)^2-\sum_{i=1}^{n-1}c_{p,i}^2(n-i-\frac{1}{2})\end{aligned}$$

只考虑p是小颜色的情况。可以在$O(\frac{N^2}{S})$内解决。

剩下的一种情况就是p,q都是小颜色的情况。我们发现如果A_l,A_r是相同的小颜色，那么不同区间(l,r)的个数是$O(NS)$的。把所有区间(i,k)看成平面上的点，一对(j,l)的答案可以看成一个矩形内的点数。使用树状数组离线扫描可以做到时间复杂度$O(NS\log N)$求出这一部分的答案。

考虑减去$A_i=A_j$的情况，对于每种颜色，如果它出现了x次，那么减去$\binom{x}{4}$。

我们令$O(NS\log N=\frac{N^2}{S})$，得$S=O(\sqrt{\frac{N}{\log N}})$，复杂度$O(N\sqrt{N\log N})$。注意卡常数。

时间复杂度：$O(N\sqrt{N\log N})$

空间复杂度：$O(N\sqrt{\frac{N}{\log N}})$

§45 LPARTY

§45.1 题目描述

对于一个只由$\{0,1,*\}$组成的长度为N的串s，它表示长度为N的01串组成的一个集合，一个01串t在该集合中当且仅当将s中的*号替换成0或1可以变成t。这个集合的代价为s中0,1的个数。

给出一个由长度为N的01串组成的集合S，求代价之和尽量小的一些集合，使得它们的并集等于S。

§45.2 数据范围

数据组数不超过120，$N\le 5$。

§45.3 解题报告

考虑预处理出合法的所有01*串，每一个01*串s对应一个集合$S(s)$，这个集合中的元素是输入串。输入串t属于$S(s)$当且仅当s可以代表t。每个01*串有一个代价，我们需要选出代价之和尽量小的集合，使得所有输入串都至少被一个集合包含。

这是一个经典的NP完全问题(set cover problem)，我们可以使用爆搜或随机化来解决它。我使用的是随机调整，以一个排列p作为状态，p的估价值为以下贪心算法的答案：按照p的顺序访问所有集合，如果下一个集合不是**前面集合的并**的子集，那么选这个集合；否则不选。后继状态可以用交换两个元素的方式产生。随机调整的次数可以取$X=30000$，不会TLE，效果不错。

时间复杂度：$O(T3^NX)$

空间复杂度：$O(6^N)$

§46 SEINC

§46.1 题目描述

有一个包含n个整数的序列A和一个包含n个整数的序列B。一次操作可以选一对$(i,j)(1 \le i \le j \le n)$，然后对所有$i \le k \le j$的元素$A_k$增加1后模4。求让$A$变成$B$的最少操作次数。

§46.2 数据范围

数据组数不超过10，$1 \le n \le 10^5, 0 \le A_i, B_i < 4$。

§46.3 解题报告

将A, B相减，问题转化为每次对一个区间+1，将序列变成全零序列。

差分后变成，每次选$1 \le i < j \le n+1$，A_i增1，A_j减1，将序列变成全零序列。

如果不是在模意义下，答案就等于A中所有数绝对值之和的一半。(注意A中所有数的和为0)

现在在模意义下，我们可以进行一系列这样的操作：选出$i < j$，将A_i增加4，A_j减少4。我们知道为了最小化绝对值之和，只有$-2, -3$这几个数可以+4，$2, 3$可以-4。对于每一个$2, 3$找出前面的一个没有被用过的$-2, -3$然后执行操作即可。注意优先考虑$3, -3$。

时间复杂度：$O(TN)$

空间复杂度：$O(N)$

§47 TKCONVEX

§47.1 题目描述

给出N根木棍的长度$A_1, A_2, \ldots, A_n$，问是否可以从中选出两组木棍每组k根，组成两个非退化凸k边形。

§47.2 数据范围

$N \le 1000, k \le 10$，木棍长度不超过10^9。

§47.3 解题报告

首先需要知道，长度为$a_1, a_2, \ldots, a_k$的木棍能组成凸k边形的充要条件为

$$2\max(a_1, a_2, \ldots, a_k) < a_1 + a_2 + \cdots + a_k$$

如果只需要求一组解，可以证明如果存在解，那么木棍长度排序后一定有一个长度为k的区间为一组解。

现在要求两组解$A_{a_1}, A_{a_2}, \ldots, A_{a_k}$及$A_{b_1}, A_{b_2}, \ldots, A_{b_k}$，其中$a_i < a_{i+1}, b_i < b_{i+1}$。我们有两种情况：

- $a_k < b_1$或$b_k < a_1$。直接枚举两个区间即可，复杂度$O(n^2)$。
- 其他：枚举第一组解$A_{a_1}, A_{a_2}, \ldots, A_{a_k}$，再把它们删掉之后枚举每个区间是不是一组解即可。事实上由于$a_k - a_1 < 2k$所以只有$O(n\binom{2k-1}{k})$种方案需要枚举，而如果第一种情况没有找到解那么n不会很大。

时间复杂度：$O(n\binom{2k-1}{k} + n^2)$

空间复杂度：$O(n)$

§48 TWOROADS

§48.1 题目描述

平面上有n个点，无三点共线。求两条直线，最小化所有点的sadness值的平均值，输出这个平均值。一个点的sadness值是它到两条直线的距离的较小值的平方。

§48.2 数据范围

$3 \le N \le 100$，坐标范围$[0, 10^3]$。

§48.3 解题报告

首先考虑只有一条直线的情况。我们可以$O(N)$地得到答案，并且可以支持$O(1)$地修改并维护答案。记一条直线的时候，点集S的答案为$p(S)$。

考虑两条线的情况，两条线的两条角平分线l_1, l_2一定是互相垂直的，形成一个平面直角坐标系。一、三象限的点记为S_1，二、四象限的点记为S_2，那么答案就是$p(S_1) + p(S_2)$。这样我们枚举所有的垂直直线对然后分别计算就可以得到答案了。

注意到本质不同的垂直直线对只有$O(N^3)$种，一定是一条直线l_1被两个点确定，另一条直线l_2经过第三个点。枚举l_1经过的点，l_2沿着l_1的方向扫的时候可以维护$p(S_1) + p(S_2)$，复杂度瓶颈为排序。

下面详细地说明如何解决只有一条直线的情况：

记$S_{f(x,y)}$表示对所有点(x, y)，$f(x, y)$的和。例如S_{xy}表示$\sum_{i=1}^{n} x_i y_i$。

设这条直线方程为$Ax + By + C = 0$，其中$A^2 + B^2 = 1$。那么问题变成求下列和式的最小值：

$$\sum_{i=1}^{n}(Ax_i + By_i + C)^2 = A^2 S_{x^2} + B^2 S_{y^2} + C^2 n + 2ABS_{xy} + 2ACS_x + 2BCS_y$$

在A, B确定的情况下，取$C = -(AS_x + BS_y)/n$可以达到该式子的最小值。最小值等于

$$A^2(S_{x^2} - \frac{S_x^2}{n}) + B^2(S_{y^2} - \frac{S_y^2}{n}) + 2AB(S_{xy} - \frac{S_x S_y}{n})$$

令$p = S_{x^2} - \frac{S_x^2}{n}, q = S_{y^2} - \frac{S_y^2}{n}, r = S_{xy} - \frac{S_x S_y}{n}, A = \sin\theta, B = \cos\theta$，原问题变成最小化

$$p\sin^2\theta + q\cos^2\theta + 2r\sin\theta\cos\theta = \frac{q+p}{2} + \frac{q-p}{2}\cos 2\theta + r\sin 2\theta$$

其最小值显然是

$$\frac{p+q}{2} - \sqrt{r^2 + \left(\frac{q-p}{2}\right)^2}$$

这样求出$n, S_{x^2}, S_x, S_{y^2}, S_y, S_{xy}$就可以了。动态的问题，维护这六个值即可。

时间复杂度：$O(N^3 \log N)$

空间复杂度：$O(N)$

§49 DEVLOCK

§49.1 题目描述

给出三个正整数N, P, m，对所有$0 \le i \le m$，求出小于10^N，各位数之和小于等于i的数中，P的倍数有多少个。答案模998244353。

§49.2 数据范围

$N \le 10^9$；对于一部分数据$P \le 50, m \le 500$；对于另一部分数据$P \le 16, m \le 15000$。

§49.3 解题报告

令$f_{i,j,k}$为小于10^i的，模P为j的，数字和为k的数的个数。如果给出f_i我们可以求出f_{i+1}和f_{2i}，那么可以分治求出答案。

$$f_{i+1,(10j+x) \bmod P,k+x} += f_{i,j,k} (0 \le x \le 9) \tag{1}$$

$$f_{2i,j,k} = \sum_{0 \le c,d < P, 10^i c + d \equiv j \pmod P} \sum_{a+b=k} f_{i,c,a} f_{i,d,b} \tag{2}$$

由以上(1)式，可以在$O(BPm)$的时间内由f_i推出f_{i+1}。$(B = 10)$

考虑(2)式，首先预处理出所有$f_{i,j}$的DFT，然后$O(P^2)$枚举c, d得到$f_{2i,j}$的DFT，然后IDFT回去即可在$O(Pm(P + \log m))$的时间由f_i推出f_{2i}。

时间复杂度：$O(Pm(B + P + \log m) \log N)$

空间复杂度：$O(Pm)$

§50 MISINT2

§50.1 题目描述

Chef的弟弟喜欢学Chef说话。当然，Chef很讨厌他这样。Chef决定十分谨慎地说话。幸运的是，Chef知道他的弟弟会怎样转换Chef说的单词。他决定只使用那些在他弟弟转换之后仍然保持不变的单词。

如果Chef说一个有N个字母的单词，他的弟弟会将所有在偶数位置的字母(假定字母位置从1开始)喵到单词开头，然后剩下的字母按照出现顺序依次排列下来。例如，abdef变成beadf；cdcd变成ddcc。

Chef想知道他能用多少个长度在$[L, R]$之间的单词(每个字母小写)。他们使用在Byteland的一种古代语言，在上面的规定下，所有可能的单词都能使用。

§50.2 数据范围

数据组数不超过5，$1 \le L, R \le 10^{10}$，$R - L \le 50000$。

§50.3 解题报告

令$\tau(n)$表示n的因子个数，$p(n)$是Chef的弟弟将Chef的单词转换用的排列，$l(p)$表示排列p的轮换个数。则我们要求的就是

$$\sum_{i=L}^{R} 26^{l(p(i))}$$

我们只考虑n为偶数的情况，因为n为奇数时$l(p(n)) = l(p(n-1)) + 1$。而n为偶数意味着$p(n)_i = 2i \bmod (n+1)$。设$q(p, i)$为排列p中i所在的轮换长度，对于一个排列$p(n)$它的轮换个数显然就是

$$l(p(n)) = \sum_{i=1}^{n} \frac{1}{q(p(n), i)}$$

然后我们注意到$q(p(n), i)$就是最小的$k (k > 0)$，使得$2^k i \bmod (n+1) = i$，即$2^k \bmod \frac{n+1}{\gcd(n+1,i)} = 1$。记这个数$k$为$G(\frac{n+1}{\gcd(n+1,i)})$。

故

$$l(p(n)) = \sum_{i=1}^{n} \frac{1}{G(\frac{n+1}{\gcd(n+1,i)})} = \sum_{t|(n+1)} [t > 1] \frac{\varphi(t)}{G(t)}$$

我们需要快速求出$G(t)$，在此之前先看G函数的两个性质：

定理3. 设$(a, b) = 1$，则$G(ab) = \operatorname{lcm}(G(a), G(b))$。

定理4. 设p是质数，$k > 1$，则$G(p^k) = \begin{cases} pG(p^{k-1}) & 2^{G(p^{k-1})} \bmod p^k \ne 1 \\ G(p^{k-1}) & 2^{G(p^{k-1})} \bmod p^k = 1 \end{cases}$

这两个性质给了我们一种方法，对于一个数n，只需要知道n的质因数的G值就可以推出n的所有因数的G值。对于一个质数p，我们知道$G(p)|(p-1)$，只要将$p-1$分解质因数，就可以在$O(\log^2 p)$的时间内求出$G(p)$。

对于$[L,R]$中所有偶数i用如此方法可以求出$l(p(i))$，再求出答案即可。

下面分析复杂度：首先分解质因数的复杂度是$O((R-L)R^{\frac{1}{4}})$，是可以承受的。

(注：我写的并不是$O(n^{\frac{1}{4}})$的Pollard-rho，而是暴力质因数分解。因为Pollard-rho的实际表现不如暴力，所以实现的时候用暴力，但是分析复杂度的时候还是用Pollard-rho的复杂度)

令$D(n)=\sum_{i=1}^{n}\tau(i)$，那么我们的复杂度是求$O(D(R)-D(L-1))$次lcm的复杂度。而易得

$$D(n)=\sum_{i=1}^{n}\lfloor\frac{n}{i}\rfloor$$

该式子的一个推论就是

$$n\ln n-2\sqrt{n}\le D(n)\le n\ln n$$

即$D(R)-D(L-1)=O((R-L)\ln R+\sqrt{R})$。

最后一部分复杂度是对于$L\le n\le R$，n为偶数的所有n，对$n+1$的质因数求G值。它不会超过$O((R-L)(\log^2 R+R^{\frac{1}{4}}))$。

本题略卡常数，注意实现。

时间复杂度：$O(T(R-L)\log R(\log R+R^{\frac{1}{4}})+T\sqrt{R})$

空间复杂度：$O(\max_{i=L}^{R}\tau(i))$

§51 DAGCH

§51.1 题目描述

给出一个N个点，M条边的有向图，1号点可以达到所有点。**按照dfs进入时刻对所有点进行了编号**。一个节点x是另一个节点y的**supreme vertex**，如果存在一条有向路$x=v_0,v_1,\ldots,v_k=y$，满足$\forall 0\le i\le k, x\le y\le v_i$。一个节点$w$的**superior vertex**是其**supreme vertex**中，编号最小的。

有Q个询问，每次给出一个节点v，询问有多少个节点将其视为**superior vertex**。

§51.2 数据范围

数据组数不超过10，$2\le N\le 10^5$，$N-1\le M\le 2\times 10^5$，$1\le Q\le 10^5$。

§51.3 解题报告

如果你稍微熟悉Lengauer和Tarjan发明的Dominator-tree的求法的话，应该可以发现所谓的**superior vertex**就是一个点的**半必经点**。

直接使用Dominator-Tree的求法即可。

时间复杂度：$O(M\log N)$

空间复杂度：$O(M)$

§52 GRAPHCNT

§52.1 题目描述

给定一个N个点M条边的有向图，求无序对(X,Y)的个数，其中(X,Y)满足存在一条从点1到点X的路径，和一条从点1到点Y的路径，且两条路径除了点1以外没有公共点。

§52.2 数据范围

$1\le N\le 10^5, 0\le M\le 5\times 10^5$。

§52.3 解题报告

对于一对(X,Y)，我们考虑如何判断是否存在这样的两条路径。

考虑网络流建模，将每个点x拆成x'和x''，源点为$1''$，对于每条有向边(a,b)，在a''和b'之间连一条容量为$+\infty$的弧；对于每一个点a，在a'和a''之间连一条容量为1的弧。X''与Y''向汇点连边。存在符合题意的路径当且仅当最大流流量为2。

换一种思路，最大流等于最小割，而最小割割边只能是a'向a''连的容量为1的弧。割掉这样一条边相当于在原图中删除点a。故我们有

定理5. (X,Y)对答案有贡献，当且仅当割掉任意一个不是1的点之后都存在点1到点X的路径，或者存在点1到点Y的路径。换句话说，就是$dom(x)\cap dom(y)=\{1\}$，其中$dom(a)$表示从1走到a必须经过的点的集合。

去掉图中1不能到达的部分，求出该图以1为根的Dominator-Tree，即询问多少对点的LCA为1。统计点1的每个儿子子树大小即可。

时间复杂度：$O(M\log N+N)$

空间复杂度：$O(M+N)$

§53 CLONES

§53.1 题目描述

令$B=\{0,1\}$，boolean function是定义在B^n，值域为B的函数。有四种集合：

- Z是满足$f(0,0,\ldots,0)=0$的函数组成的集合；
- P是满足$f(1,1,\ldots,1)=1$的函数组成的集合；
- D是满足$\forall x_1,x_2,\ldots,x_n,!f(x_1,x_2,\ldots,x_n)=f(!x_1,!x_2,\ldots,!x_n)$的函数组成的集合；(!表示非)
- A表示这样的函数f组成的集合：对于任意$a_1,a_2,\ldots,a_n,b_1,b_2,\ldots,b_n,c,d$和位置$i$，如果$f(a_1,a_2,\ldots,c,\ldots,a_n)=f(a_1,a_2,\ldots,d,\ldots,a_n)$，那么$f(b_1,b_2,\ldots,c,\ldots,b_n)=f(b_1,b_2,\ldots,d,\ldots,b_n)$。

q组询问，每组询问用含有Z,P,D,A,v(并),^(交),!(补),\(差)和括号的表达式来描述一个集合，求这个集合的大小模1000003。

§53.2 数据范围

$n,q\le 100$，表达式长度不超过100。

§53.3 解题报告

我们用$\bar{S}$来表示S的补集；省略“取大小(||)”和“交($\cap$)”不写。例如说用$\bar{Z}PD\bar{A}$来表示在$P\cap D$中但是不在$Z\cup A$中的函数的个数。对于四种集合的每个状态(可以理解为“上面有没有横线”)我们求出其大小。一共16个数。每次询问的时候只要取出符合表达式的部分加起来即可。

首先可以发现A中的函数一定形如将$x_1,x_2,\ldots,x_n$中取出一些变量求异或和(可能最后还要异或1)。

然后可以发现：

1) $Z=P=2^{2^n-1}$
2) $D=2^{2^{n-1}}$
3) $A=2^{n+1}$
4) $ZP=Z\bar{P}=\bar{Z}P=\bar{Z}\bar{P}=2^{2^n-2}$
5) $DZP=D\bar{Z}\bar{P}=2^{2^{n-1}-1}$
6) $DZ\bar{P}=D\bar{Z}P=0$
7) $AZP=A\bar{Z}P=AZ\bar{P}=A\bar{Z}\bar{P}=2^{n-1}$
8) $ADZP=AD\bar{Z}\bar{P}=2^{n-1}$

9) 根据4),5),6)可以求出$\bar{D}ZP,\bar{D}Z\bar{P},\bar{D}\bar{Z}P,\bar{D}\bar{Z}\bar{P}$。
10) 根据5),8)可以求出$\bar{A}DZP,\bar{A}D\bar{Z}\bar{P}$。
11) 根据7),8)可以求出$A\bar{D}ZP,A\bar{D}Z\bar{P},A\bar{D}\bar{Z}P,A\bar{D}\bar{Z}\bar{P}$。
12) 根据9),11)可以求出$\bar{A}\bar{D}ZP,\bar{A}\bar{D}Z\bar{P},\bar{A}\bar{D}\bar{Z}P,\bar{A}\bar{D}\bar{Z}\bar{P}$。

对于一组询问，枚举$ADZP$可能的情况一共2^4种，代入表达式计算，如果为真那么将其方案数计入答案。本题暴力计算表达式可以过，复杂度$O(L^2)$。

时间复杂度：$O(2^4L^2q)$

空间复杂度：$O(2^4+L)$

§54 LEBOXES

§54.1 题目描述

利沃夫动物园的小象有n个盒子，每个盒子里有一些钱或者钻石。他不知道每个盒子里具体是什么，但他知道如果打开第i个盒子，他有$\frac{P_i}{100}$的概率能获得V_i美元的钱，否则能获得一个钻石。

现在有m个物品，第j个物品需要花费恰好C_j美元的钱和D_j个钻石。小象非常聪明，当他获得了一定量的钱和钻石后，他总会买尽可能多的物品。注意，每个物品只能购买一次。现在请你帮助小象计算出，当他打开所有的盒子后，他期望能够买到的物品个数。

§54.2 数据范围

数据组数不超过5，$2\le n\le 30$，$1\le m\le 30$，$1\le V_i,C_j\le 10^7$，$0\le D_i\le 30$，$0\le P_i\le 100$。

§54.3 解题报告

首先预处理出$f_{i,j}$表示有i个钻石，想买j个物品至少需要多少钱。这个可以用$O(nm^2)$的dp解决。

然后，meet in the middle。

预处理出前r个盒子的所有可能情况，即2^r个三元组(d,c,p)，分别表示所得的钻石数、钱数以及发生这种情况的概率。记这些三元组的集合为S_1。

考虑后$n-r$个盒子的某一种情况(d,c,p)，我们要求S_1中的所有三元组与该情况加起来对答案的贡献，即

$$\sum_{(d',c',p')\in S_1} pp'\max\{j:f_{c+c',j}\le d+d'\}$$

枚举$c+c'$和j，可以确定d'的范围，问题就变成求对于指定c'，d'在一定范围的三元组，p'的和。对于一个c'将所有$c=c'$的三元组以d为关键字进行排序，对p作前缀和，即可在$O(\log 2^r)=O(r)$的时间复杂度内回答这种询问。

复杂度$O(r2^r+2^{n-r}rm)$，取$r=\frac{\log_2 m+n}{2}$可以取得最优复杂度。

时间复杂度：$O(nm^2+(\log m+n)\sqrt{m2^n})$

空间复杂度：$O(nm+\sqrt{m2^n})$

§55 QTREE6

§55.1 题目描述

给出一棵n个点的树，每个节点有颜色(黑或白)，支持m个操作：

- 0 u：求点u所在的连通块的大小，要求只经过两端是同色点的边。
- 1 u：改变点u的颜色。

§55.2 数据范围

$1\le n,m\le 10^5$。

§55.3 解题报告

令b_i表示如果强制把i染成黑色，那么只考虑i的子树，i所在连通块的大小。w_i类似。

修改的时候，假设把点u从黑色改成白色。首先只有u的祖先的b, w值会被改变。对于u的一个祖先a，如果u到a的路径(不包含u, a)上的点都是白色，那么a的w值加上u的w值；如果u到a的路径(不包含u, a)上的点都是黑色，那么a的b值减去u的b值。

询问的时候，找到u的祖先中深度最小的点p，使得u到p的路径(包含u, p)上的点同色。如果u是黑色那么询问b_p；否则询问w_p。

用树链剖分维护一下就好。需要维护的信息有：b，w，以及一段路径的颜色(黑/白/混杂)。

时间复杂度：$O(n + m\log^2 n)$

空间复杂度：$O(n)$

§56 challenge COMPLEXT

§56.1 题目描述

给出一个$n \times n$网格图，每条边有复数边权。求一个生成树使得树边上的权值和的模最大。

§56.2 数据范围

$n = 20$，其他数的绝对值不超过100。

§56.3 解题报告

如果我们知道了最后的权值和Z，那么我们做生成树的时候，把每个复数看成一个向量，以边权与Z的点积为权值做实数权最大生成树即可。

随机Z的方向(只需要随机一个角度θ，然后令$Z = \cos\theta + \mathrm{i}\sin\theta$即可)，做最大生成树，重复足够多次即可得到较优解。

时间复杂度：$O(n^2 + count \times n^2)$

空间复杂度：$O(n^2)$

§57 COT5

§57.1 题目描述

一个treap是这样的一棵树：每个点有两个权值(key,weight)，树是关于key的二叉查找树，也是关于weight的堆(本题中是大根堆)。维护一个treap，支持三种操作：

- $0\ k\ w$：在treap中插入一个新的节点，其$\text{key} = k, \text{weight} = w$；
- $1\ k$：将treap中$\text{key} = k$的点删除；
- $2\ k_u\ k_v$：找出treap中$\text{key} = k_u, k_v$的两点，输出其距离。

§57.2 数据范围

操作数$N \le 200000$。$0 < \text{key}, \text{weight} < 2^{32}$。树中不会存在相同key或相同weight 的节点。

§57.3 解题报告

要支持求树上两点之间的距离，只要支持求两点的LCA和一个点的深度。

对于两个点(k_1, w_1)和(k_2, w_2)，它们的LCA是key在k_1, k_2之间的点中，weight最大的那个。

考虑求一个点的深度，我们把点按照key升序排序，把weight依次写下：$w_1, w_2, \ldots, w_n$。如果求w_i的深度，那么维护一个变量$tmp = w_i$，从w_i开始往右走，每走到一个w_j，如果$w_j > tmp$，令$tmp = w_j$。则tmp的变化次数就是根走到w_i的路径中，往左走的次数。同理可得往右走的次数，即得到了深度。

LCA是很好维护的，但是深度则不太好维护。我们可以用线段树维护这个序列。

令$l(x), r(x)$分别表示x的左、右儿子，$m(x)$表示x内的最大值，$Q(W, x)$表示以$tmp = W$进入点x，在x内tmp的变化次数。这样询问$Q(W, x)$的时候：

- 如果$m(l(x)) < W$，那么返回$Q(W, r(x))$；
- 否则返回$Q(W, l(x)) + Q(m(l(x)), r(x))$。

这样我们对于线段树的点x，维护$Q(m(l(x)), r(x))$即可。

一次询问复杂度是$O(\log N)$的，而一次修改需要更新$O(\log N)$个点，需要做$O(\log N)$次询问，故复杂度是$O(\log^2 N)$的。不过常数较小。

对于插入、删除操作，离线搞出所有出现过的点，不在树中的点权值设为$-\infty$(不同的点权值不同，且都小于0)即可。

时间复杂度：$O(N \log^2 N)$

空间复杂度：$O(N)$

§58 FINDSEQ

§58.1 题目描述

给出一个长度为N的数组A和一个$1 \sim 5$的排列p，下标均从0开始。求五个整数i_0, i_1, i_2, i_3, i_4，满足

- $0 \le i_0 < i_1 < i_2 < i_3 < i_4 < N$；
- $A_{i_0}, A_{i_1}, A_{i_2}, A_{i_3}, A_{i_4}$互不相等；
- 对任意$0 \le j < 5$，A_{i_j}是$A_{i_0}, A_{i_1}, A_{i_2}, A_{i_3}, A_{i_4}$中第$p_j$小的。

给出一组解或说明无解。

§58.2 数据范围

数据组数不超过60，$N \le 1000, |A_i| \le 10^9$。

§58.3 解题报告

枚举i_1, i_3。那么考虑$A_{i_0}, A_{i_2}, A_{i_4}$中最小的那个数所在的区间($[0, i_1), (i_1, i_3), (i_3, N)$中的某一个)和最大的数所在的区间。我们知道这两个数与A_{i_1}, A_{i_3}之间的大小关系，那么我们可以贪心地求出这两个数。例如，对于最小的数我们知道它大于$A_{i_1}, A_{i_3}, 0$中的某个数，在它对应的范围内求出大于该数的最小数即可。

接下来确定$A_{i_0}, A_{i_2}, A_{i_4}$的中位数，我们知道它的范围$(l, r)$和它所在的区间$[L, R]$，求出$[L, R]$中大于$l$的最小数，并检查它是否小于$r$即可。

我们用三个平衡树维护$[0, i_1), (i_1, i_3), (i_3, N)$中的数，支持查询前趋/后继即可。当然也可以离散化后使用线段树，常数更小。

时间复杂度：$O(TN^2 \log N)$

空间复杂度：$O(N)$

§59 CONNECT

§59.1 题目描述

给出$N \times M$的矩阵，每个格子上有一个数。选出一个四连通块，使得连通块中不含-1，且至少有K种不同的正数。每一个格子都有被选的代价。你需要最小化选格子的代价。

§59.2 数据范围

$1 \le N, M \le 15, 1 \le K \le 7$，格子上的数在$[-1, NM]$内，代价在$[1, 100000]$内。

§59.3 解题报告

假设格子上的数在$[-1,K]$内。那么这是经典的斯坦纳树问题。令$f_{i,j,s}$表示经过点(i,j)，且内部的颜色集合为s的四连通块的最小权值，转移如下：

- $f_{i,j,s_1}+f_{i,j,s_2}-cost_{i,j}\rightarrow f_{i,j,s_1\cup s_2}$
- $f_{i,j,s_1}+1\rightarrow f_{k,l,s_1}$，其中$(i,j)$与$(k,l)$相邻。

复杂度可以看成$O(NM3^K)$。

考虑数的个数变多了，设有V种不同的数，我们用随机函数f将$1\dots V$的数映射到$1\dots K$，对映射之后的矩阵跑斯坦纳树。显然答案不会小于原题的最优解。

设原题的答案中，最优解对应的数为$A_1,A_2,\dots,A_k$，我们发现只要$f(A_1),f(A_2),\dots,f(A_k)$两两不相等那么答案就等于原题的最优解。这个概率是$\frac{K!}{K^K}$。重复$X=370$次即可以很高概率得到最优解。

时间复杂度：$O(XNM3^K)$

空间复杂度：$O(NM2^K)$

§60 SHORT2

§60.1 题目描述

给定p(一个质数)，问有多少对$a,b(a>p,b>p)$满足ab被$(a-p)(b-p)$整除。

§60.2 数据范围

数据组数不超过5，$1<p<10^{12}$。

§60.3 解题报告

我们称满足条件$ab\mid(a+p)(b+p)$的数对(a,b)为“好的”。令

1) X表示满足$p\mid a\wedge p\mid b$的好的数对个数。
2) Y表示满足$p\nmid a\wedge p\mid b$的好的数对个数。
3) Z表示满足$p\nmid a\wedge p\nmid b$的好的数对个数。

我们要求的就是$X+2Y+Z$。

首先，如果$p\mid a\wedge p\mid b$，那么令$a=a_0p,b=b_0p$，即$a_0b_0\mid(a_0+1)(b_0+1)$。容易得到$X=5$。

考虑求Y的值。此时令$b=b_0p$，就有$(ab_0)\mid(a+b_0p+p)$。令$a+b_0p+p=kab_0$，可得$b=\frac{a+p}{ka-p}$。令$ka-p=d$，我们可以证明$p\nmid d$，又$d+p=ka$，$a+p=bd$，故$ad\mid(a+p)(d+p)$。即(a,d)是好的。故由一个2)中的数对(a,b)可以得到一个3)中的数对(a,d)。

而给出3)中的数对(a,d)，由于$ad\mid(a+p)(d+p),\gcd(a,a+p)=1$，所以$d\mid(a+p),a\mid(d+p)$。令$b=\frac{p(a+p)}{d}$。要证明$ab\mid(a+p)(b+p)$只需要证(省略推式子)$a\mid(a+p+d)$，而这是显然的。于是我们由一个3)中的数对得到了一个2)中的数对。而这个映射明显是上面那个映射的逆。这表明，$Y=Z$。

现在考虑求Z。这意味着$ab\mid(a+b+p)$。如果$a=b$，显然$a=b=1$。否则，求出$a<b$的好的二元组对数乘以2即可。而$ab\le(a+b+p)\Rightarrow(a-1)(b-1)\le p+1\Rightarrow a\le\sqrt{p+1}+1$。

由$a+b+p=kab$可以得到$b=\frac{a+p}{ka-1}$，令$d=ka-1$，即需要寻找这样的数对(a,d)数目：

I $p\nmid a$;
II $d\mid(a+p)$;
III $a\mid(d+1)$;
IV $b=\frac{a+p}{d}>a$。

令$W=\sqrt{p+\sqrt{p+1}+1}$为$\sqrt{a+p}$的上界。如果$d\le W$，那么直接枚举d，而由II，$a\equiv -p(\bmod d)$，由III，$a\le d+1$。故可能的a数量是$O(1)$的。

如果$d > \sqrt{a+p}$，那么枚举b，由于$0 < a = bd - p < b$，所以d只有一种取值。

故我们证明了可能的(a, d)数目是$O(\sqrt{p})$的，逐一代入验证即可。

时间复杂度：$O(T\sqrt{p})$

空间复杂度：$O(1)$

§61 SEAORD

§61.1 题目描述

Sereja有N个程序，每个程序都需要在两台电脑上分别运行。第i个程序需要在第一台电脑上运行A_i秒，在第二台电脑上运行B_i秒。一台电脑不能同时运行两个程序，一个程序也不能同时在两台电脑上运行。

Sereja需要用最少的时间完成所有程序在两台电脑上运行的任务，请你帮帮他。

§61.2 数据范围

$1 \le N \le 10000$。多组数据，$\sum N \le 2 \times 10^5$。

§61.3 解题报告

答案至少是$\max(\max_{i=1}^N A_i + B_i, \sum_{i=1}^N A_i, \sum_{i=1}^N B_i)$。可以证明取得到这个下界。

记$T = \max(\sum_{i=1}^N A_i, \sum_{i=1}^N B_i)$。如果存在$i$使得$A_i + B_i \ge t$，那么可以直接构造答案。否则，我们随机每台电脑运行所有程序的顺序，随机到一个合法解即为答案。事实上可行的顺序很多，所以可以很轻松地得到答案。

时间复杂度：$O(TN \times \text{玄学})$

空间复杂度：$O(N)$

§62 RIVPILE

§62.1 题目描述

有一条河，可以用集合$\{(x, y) : 0 \le y \le W\}$表示。集合中有$N$个木桩，它们的位置已知。有$M$种圆环，第$i$种的半径是$R_i$，价格是$C_i$，同一种圆环可以被使用多次。请选择一些木桩，给这些木桩安上圆环，使得存在一条从某个$y = 0$的点到某个$y = W$的点的曲线，曲线上任意一点属于至少一个圆环。最小化这些圆环的价格之和。

§62.2 数据范围

数据组数不超过10，$1 \le N, M \le 250$。

§62.3 解题报告

考虑建立NM个点$(i, j)(1 \le i \le N, 1 \le j \le M)$，点$(i, j)$表示木桩$i$的圆环是第$j$种。对于$(i, j)$和$(k, l)$，如果木桩$i$和木桩$k$之间的距离不超过$R_j + R_l$，那么可以连一条有向边，代价为$C_l$。

建立源点S和汇点T，如果$Y_i \le R_j$那么S向(i, j)连有向边，代价为C_j；如果$M - Y_i \le R_j$那么(i, j)向T连有向边，代价为0。于是答案变成了S到T的最短路径长度。

该图的边数是$O((NM)^2)$的，考虑优化连边。我们去掉无用的圆环后，将圆环按照R排序，即$R_1 < R_2 < \cdots < R_M$。可以认为$C_1 < C_2 < \cdots < C_M$。如果$l < M$且存在有向边$(i, j) \to (k, l)$，那么显然存在有向边$(i, j) \to (k, l+1)$，且两条边代价之差为$C_{l+1} - C_l$。

对于每个点(i, j)，如果$j < M$那么将其与$(i, j+1)$之间连一条代价为$C_{j+1} - C_j$的有向边。对于(i, j)和点k，找出最小的l使得$R_l + R_j \ge \sqrt{(X_i - X_k)^2 + (Y_i - Y_k)^2}$。将$(i, j)$和$(k, l)$之间连代价为$C_l$的有向边。注意到$i, k$确定了之后，$l$关于$j$是单调的。这样可以在$O(N^2M)$的时间内完成连边。

这样图的边数变成了$O(N^2M)$。求最短路的dijkstra算法的复杂度是$O(N^2M(\log N+\log M))$。使劲卡常数可以通过本题。

时间复杂度：$O(N^2M(\log N+\log M))$

空间复杂度：$O(N^2M)$

§63 CUCUMBER

§63.1 题目描述

已知B个$N\times N$的矩阵$Q_1,Q_2,\dots,Q_B$。对于每对$a,b(1\le a<b\le n)$，定义$C_{a,b}$是一个$N\times N$的矩阵，满足

$$C_{a,b}[i][j]=\sum_{k=1}^{N}Q_a[i][k]Q_B[j][k]$$

求满足如下条件的a,b的个数：令$CNT_{a,b}$表示$1\sim n$的排列p中，满足$C_{a,b}[i][p_i](1\le i\le N)$中至少有一个奇数的排列$p$的数目。条件为$CNT_{a,b}$为奇数。

§63.2 数据范围

$1\le N\le 60$。多组数据，所有数据中B的和不超过8000。

§63.3 解题报告

对于$N=1$的情况，求$Q_a[1][1]Q_b[1][1]$是奇数的方案数即可。时间复杂度$O(B^2)$。

对于$N>1$的情况，$N!$为偶数。我们构造矩阵$C'_{a,b}[i][j]=[C_{a,b}[i][j]\text{为偶数}]$，则$CNT_{a,b}$为奇数当且仅当$C'_{a,b}$的积和式为奇数。而在模2意义下积和式与行列式是相等的。

我们可以将矩阵C'写成两个矩阵相乘的形式。考虑在每个Q_i后面添加一列，且令每个$Q_{i,N+1}=1$。那么$C'_{a,b}=Q_aQ_b^T$。

根据Cauchy-Binet公式，

$$\det(C'_{a,b})=\sum_{i=1}^{N+1}\det(Q_{a,i})\det(Q_{b,i})$$

其中$Q_{a,i}$表示矩阵Q_a去掉第i列的行列式。

对于每个矩阵Q_i我们记$P_{i,j}$为$\det(Q_{i,j})$的奇偶性，那么答案就是

$$\sum_{1\le a<b\le n}(\sum_{i=1}^{N+1}P_{a,i}P_{b,i}\mod 2)$$

如果压位的话可以在$O(B^2)$的时间内求得答案。

现在问题变成了求$P_{i,j}$。在模2意义下考虑，首先如果Q_i不满秩，那么所有的$P_{i,j}$都是0；否则找出N列使它们线性无关，记剩下来一列为j，它是一些列的异或和。对于这些列(包括j)来说，去掉它们之后矩阵行列式仍然是1；去掉其他列之后矩阵行列式为0。

使用压位的技巧可以在$O(N^2)$的时间复杂度内处理出一个矩阵的$P_{i,j}$。

时间复杂度：$O(B(N^2+B))$

空间复杂度：$O(BN^2)$

§64 REALSET

§64.1 题目描述

给出一个循环矩阵，问其是否满秩。

§64.2 数据范围

数据组数不超过100，矩阵大小不超过30000，元素绝对值不超过1000，$\sum n\le 1.5\times 10^5$。

§64.3 解题报告

令循环矩阵的第一行为$(a_0, a_1, \ldots, a_{n-1})$，定义$f(z) = \sum_{i=0}^{n-1} a_i z^i$，那么循环矩阵的行列式为$\prod_{p=0}^{n-1} f(\omega_p)$。其中$\omega_p = e^{\frac{2\pi \mathrm{i} p}{n}}$为单位根。

该行列式为0当且仅当某个$x - \omega_p$整除$f(x)$。

我们考虑**分圆多项式**$\Phi_n(x) = \prod_{(i,n)=1}(x - \omega_p)$。我们只需要知道是否存在$k|n$使$\Phi_k(x)$整除$f(x)$。可以证明，如果$k$有$v$个质因子$p_1, p_2, \ldots, p_v$，那么$\Phi_k(x)$整除$f(x)$当且仅当$x^k - 1$整除$f(x)\prod_{i=1}^{v}(x^{k/p_i} - 1)$。在模$x^k - 1$意义下进行，那么我们可以把下标模$k$相同的系数加在一起，乘以$x^{k/p_i} - 1$就是$O(k)$的了。

令$\tau(x)$为x的因数个数，对每个因数都做一遍上述过程。

时间复杂度：$O(n\tau(n)\log n)$

空间复杂度：$O(n)$

§65 CUSTPRIM

§65.1 题目描述

定义一个Payton number(PN)是一个三元组(a, b, c)，其中$a, b \in \mathbb{Z}, c \in \{11, 24\}$。其乘法如下：

```
def multiply((a1,b1,c1), (a2,b2,c2)):
    s = (a1a2 + b1b2 + c1c2) + (a1b2 + b1a2) + (c1 + c2)
    t = floor[s/2] + 16(c1 + c2) - c1c2
    A = (t - 2(a1b2 + b1a2) - (a1c2 + c1a2) + 33(a1 + a2) + (b1b2 - a1a2))
    B = (t - 5(a1b2 + b1a2) - (c1b2 + b1c2) + 33(b1 + b2) + (2b1b2 + 4a1a2))
    if s is even:
        return (A-540,B-540,24)
    else:
        return (A-533,B-533,11)
```

用$\mathbb{PN}$表示PN组成的集合。

zero指这样的PN z：$\forall x \in \mathbb{PN}, xz = z$。存在唯一的zero $z = (11, 11, 11)$。

unity指这样的PN u：$\forall x \in \mathbb{PN}, xu = x$。存在唯一的unity $u = (4, 4, 24)$。

x**整除**y，意味着$\exists z \in \mathbb{PN}, xz = y$。

unit指能够整除unity的所有PN。

质数指这样的PN p：它不是zero或unit，也不能写成两个非zero、非unit的PN的乘积。

给出一个PN (a, b, c)，判断它是不是质数。

§65.2 数据范围

数据组数不超过10^4，$|a|, |b| \le 10^7, c \in \{11, 24\}$。

§65.3 解题报告

这是一道结论题。结论如下：

Payton numbers对应的域与$\mathbb{Z}[\omega]$同构，其中$\omega^2 = \omega - 3$。(a, b, c)对应于$(33 - 2a + c) + (b - a)\omega$。对于$x + y\omega$，它是质数当且仅当以下两个条件中的一个被满足：

- $y = 0$，且$|x|$是质数，且要么$|x| = 2$，要么$|x| \ne 11$且$(-11)^{\frac{|x|-1}{2}} \equiv -1 \pmod{|x|}$。
- $y \ne 0$，且$Nx = x^2 + xy + 3y^2$是质数。

使用Miller-Rabin判断质数，可以做到$O(k(\log|a| + \log|b|))$，其中k是Miller-Rabin的实验次数。

时间复杂度：$O(Tk(\log|a| + \log|b|))$

空间复杂度：$O(1)$

§66 SPMATRIX

§66.1 题目描述

对于一个N行N列的矩阵A，如果满足如下条件，那么我们称它为特殊的：

- $\forall 1 \le x \le N, A_{x,x} = 0$；
- $\forall 1 \le x < y \le N, A_{x,y} = A_{y,x} > 0$；
- $\forall 1 \le x, y, z \le N, A_{x,y} \le \max(A_{x,z}, A_{z,y})$；
- $\forall 1 \le x < y \le N, A_{x,y} \in \{1, 2, \ldots, N-2\}$；
- $\forall k \in \{1, 2, \ldots, N-2\}, \exists x, y \in \{1, 2, \ldots, N\}, A_{x,y} = k$。

给出N，求特殊矩阵的个数模10^9+7。

§66.2 数据范围

数据组数不超过10^5，$3 \le N \le 10^7$。

§66.3 解题报告

数列可以在`oeis.org/A059355`找到。

答案是

$$\frac{N!(N-1)!}{2^{N-1}}\left(\frac{N}{2} - \frac{2 + \sum_{i=1}^{N-1}\frac{1}{i}}{3}\right)$$

预处理阶乘、$\frac{1}{2}$的幂、$\sum_{i=1}^{N}\frac{1}{i}$在模$10^9+7$下的值，即可在$O(1)$的时间内求出一个询问的答案。

时间复杂度：$O(N+T)$

空间复杂度：$O(N)$

§67 MINESREV

§67.1 题目描述

你一定玩过著名的电脑游戏－－扫雷，以防你没玩过，这里有一个简短的对游戏规则的描述。这个游戏是在R行C列的矩形网格中进行。有一些方块含有“雷”，而剩下的方块是空的。一开始所有的方块都是关闭的，也就是说你不知道它是否有“雷”。玩家需要去打开所有的空方块，并且不能打开任何一个含“雷”的方块。如果你点击一个方块，就能打开它。如果你打开了一个空方块，它会告诉你一个数字，表示这个方块周围8个方块中含“雷”方块的数目。如果这个方块周围没有雷，所有周围方块也都会被自动打开，并可能重复这一个过程。通过这种方式，你可以通过仅仅一次点击打开一个很大的无“雷”区域。

就像题目名字那样，你将会处理一个扫雷的逆过程。开始时所有的方块都是打开的，有的含“雷”，剩下的不含。玩家的目标是尽快关掉所有的方块。有两种关掉方块的方式：你可以通过一次点击来关闭一个方块(可以关闭有“雷”的方块)，或者点击其他的方块顺便让这个方块关闭。

点击一个方块C_1以后，按照正常扫雷规则，所有原本可能和C_1同时打开的方块会全部关闭。

换句话说，令$S_{x,y}$表示在正常扫雷中打开(x,y)后会自动打开的所有其他方块的集合，点击一个方块C_1以后，所有和C_1在一个公共集合$S_{x,y}$中的C_2方块会被关闭。注意到是“点击”一个方块，所以和扫雷游戏不同，这个过程不会重复。现在要你求出至少点击多少次，可以关闭所有的方块。

§67.2 数据范围

数据组数不超过50，$1 \le R, C \le 50$。

§67.3 解题报告

首先点击所有的雷。

对于一个空格子，如果与它8连通的格子都不是雷，那么称它为1类格子，否则称为2类格子。

所有1类格子按照八连通组成了若干连通块(下面的"连通块"专指这种连通块)。如果点击一个1类格子，那么会关闭它所在的连通块的所有格子，以及与这个连通块八连通的所有2类格子。如果点击一个2类格子，那么会关闭与它八连通的所有1类连通块，以及与这些连通块八连通的所有2类格子。

然后我们点击所有的**不与任意连通块八连通的2类格子**。因为这是唯一的关闭它们的方法。剩下的2类格子都可以在点击某个连通块的时候被连带着关闭掉。

考虑点击一个1类格子，我们发现这个操作不优于点击连通块周围的2类格子。即，我们只需要点击2类格子。而对于一个2类格子，我们知道点击它可以关闭哪些连通块(与它八连通的连通块)。我们的目标是用尽可能少的点击数关闭所有连通块。

还有一个性质：一个2类格子与至多两个连通块八连通。于是对于每个2类格子，如果它与两个连通块八连通，我们在这两个连通块之间连一条边，用连通块的数目减去图的最大匹配就是答案。

使用带花树算法解决图的最大匹配。

时间复杂度：$O(TR^3C^3)$

空间复杂度：$O(R^2C^2)$

§68 TICKETS

§68.1 题目描述

对于一个有n个点、m条边的无向图(V,E)，求最大的数k，使得无论如何选出k条边$E_s=\{e_1,e_2,\ldots,e_k\}$，都存在一个$E_s\to V$的映射$f$，满足

- $\forall e\in E_s$，$f(e)$是e的一个端点；
- $\forall e_1,e_2\in E_s$，若$e_1\neq e_2$，那么$f(e_1)\neq f(e_2)$。

§68.2 数据范围

数据组数不超过15，$n\le 200$，$m\le 500$。不存在自环但是可能有重边。

§68.3 解题报告

我们证明

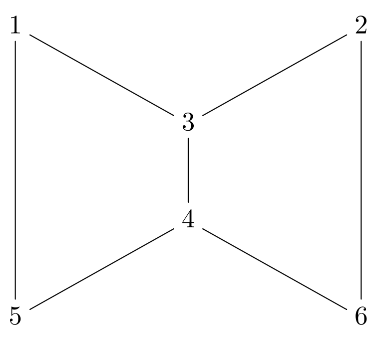
定理6. *边集E_s是可以选的(a)，当且仅当对于其每一个连通块，边数不超过点数(b)。*

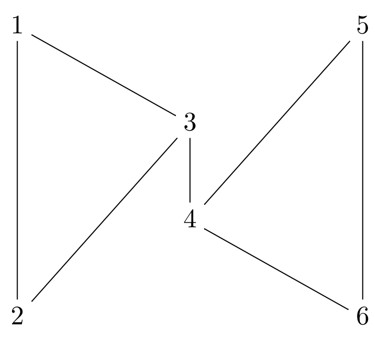
Proof. (a)⇒(b)：E_s可以选⇒每一个连通块可以选⇒(b)。

(b)⇒(a)：对于一个连通块，如果边数小于点数，那么该连通块是一棵树，任意选一个根之后，每条边的f表示其深度低的节点。如果边数等于点数，我们找出一条环边e，令$f(e)$为根即可。

现在问题变成：选出最少的边，使得它们处于同一个连通块，且边数大于连通块的点数。输出选的边的个数减1。

那么选出的边正好组成两个环。根据两个环是否相交分两种情况讨论，分别是这样的：





左图表示环相交，右图表示环不交。

对于左图情况，枚举点3和点4，求点3到点4的前三短的路径。枚举路径中点4前面的点跑bfs最短路即可。总复杂度$O(n^2m)$。

对于右图情况，枚举点3，以其为根做一棵bfs树，对于每一条非树边(x,y)求出$L=lca(x,y)$和$w=dist(3,L)+dist(L,x)+dist(L,y)$，取$w$最小的两个环。暴力的话复杂度$O(n^2m)$。

时间复杂度：$O(Tn^2m)$

空间复杂度：$O(m)$

§69 QPOINT

§69.1 题目描述

平面上有N个多边形，两两之间没有公共点。给出Q个询问，每次求一个点所在的多边形编号。如果不在任何一个多边形内，输出-1。强制在线。

§69.2 数据范围

$1\le N\le 100000, 1\le Q\le 10^5$,多边形点数之和$K$不超过$3\times10^5$，所有坐标在$[0,10^9]$内。

§69.3 解题报告

考虑射线法，从询问点出发射一条水平向右的射线，碰到的第1条非水平的边决定了其所在多边形。射线经过边的端点也算碰到。

首先考虑预处理出碰到每条边的意义。水平边直接无视；对于其他边，如果其左侧在多边形内，那么代表这个点在对应多边形内；否则代表这个点不属于任何一个多边形。对于一个多边形，我们根据有向面积可以知道它的输入顺序是顺时针还是逆时针；对于一个顺时针的多边形上的边$(x_1,y_1)\to(x_2,y_2)$，其左边在多边形中当且仅当$y_1>y_2$。

接下来我们用一根横着的扫描线，从$y=+\infty$往下扫，每碰到一条边的上端点就把这条边加入平衡树，碰到下端点就删除这条边。由于不存在相交的边，所以可以用左右顺序来作为平衡树的序。将这个平衡树可持久化。

对于每一个询问点，我们通过其y坐标可以知道它被扫到时，平衡树处于哪一个历史版本。如果处于两个历史版本的交集那么取下面那个。接下来在平衡树中询问其后面一条边，读取出该边的信息即可回答询问。

注意：判断相等的时候要用相对误差法，即$(1-\epsilon)y<x<(1+\epsilon)y$；点在多边形边上时要特判。这题卡精度，$\epsilon$最好设为$10^{-18}$或者更小。

时间复杂度：$O(K\log K+Q\log K)$

空间复杂度：$O(K\log K)$

§70 PUSHFLOW

§70.1 题目描述

给出一个n个点m条边的仙人掌，处理以下两种操作共q个：

- $0\ S\ T$：求仙人掌以S为源点，T为汇点时的最大流量；
- $1\ X\ NEW\_CAPACITY$：将编号为X的边的容量修改为$NEW_CAPACITY$。

§70.2 数据范围

$1\le n\le 100000$，$1\le m,q\le 200000$。任何时刻任一条边容量是$[1,10^9]$内的正整数，0操作中$S\ne T$。

§70.3 解题报告

考虑一次询问，首先设S到T的路径为P，一个环为Q。如果$P \cap Q \neq \emptyset$，那么Q对答案的贡献为$\min(P \cap Q) + \min(Q \backslash P)$，否则$Q$对答案的贡献为$+\infty$。在$P$中的一个桥对答案的贡献为它的权值。答案就是所有贡献的min。

dfs仙人掌弄清楚哪些边是桥，并得到一棵dfs树$Tree$。

令T_1为这样的结构：维护$Tree$上的权值，桥边的权值不变，其他边的权值为$+\infty$。询问的时候在T_1上进行询问，就可以得到所有桥的贡献。修改的时候很好维护。

对$Tree$进行轻重链剖分。对于一个环Q，设其非树边为(c,p)，其中p是c的祖先。定义$h(Q)$为从p沿着树边走到c，走到第一条轻边之前走过的所有边。

考虑所有与S到T的树上路径P有交的环Q，它们分三种情况：

- $P \cap Q$中包含轻边：路径上的轻边条数为$O(\log n)$，遍历所有轻边，访问它们所在的环，可以确定$P \cap Q$和$Q \backslash P$的边集(前者是一条路径，后者是一条路径并上一条非树边)，即可求得答案。用结构T_2来询问$Tree$上任意两点间的最小边权，T_2的维护类似T_1。
- $P \cap Q \neq h(Q)$：考虑P和Q分岔的地方，要么是起点/终点，要么是一个点，P走的是轻边，Q走的是重边。同理这种环只有$O(\log n)$个。
- 其他：构建结构T_3，维护$Tree$上的权值。如果一条边$e \in h(Q)$，边e的权值是$w_e + \min(Q \backslash h(Q))$；否则该边的权值为$+\infty$。对每一段没有被前面两种环覆盖的重链，在$T_3$中询问链上min更新答案。

我们需要维护T_3。对于每个环Q将$Q \backslash h(Q)$的所有边权丢入一个堆H_Q，修改$h(Q)$中的边的权值时直接更新T_3；修改$Q \backslash h(Q)$中的边的权值时更新这个堆，并更新所有$h(Q)$中的边在T_3中的权值。

时间复杂度：$O(n \log n + q \log^2 n)$

空间复杂度：$O(n)$

§71 challenge SEAVEC

§71.1 题目描述

给出n个k维向量$V_1, V_2, \ldots, V_n$和一个向量S，要求从中选出一些向量，使它们的和的每一维都不超过S的对应维。设选了q个向量，它们的和为V，你需要使下式尽可能大。

$$\frac{q}{1 + \sum_{i=1}^{k} S_i - V_i}$$

§71.2 数据范围

数据组数不超过10，$1 \leq NK \leq 10^5$。

§71.3 解题报告

首先我们有一个贪心算法：按照某种顺序遍历这些向量，如果某个向量能被加入答案那么加入。根据该算法可以得到一个初始解。

然后进行退火、爬山之类的随机调整。对于一个解S我们按照如下方式产生相邻解：

每次随机一个向量，如果它在S中那么删掉它；否则如果它可以被加入那么加入；否则重新随机。

该算法配合卡时可以取得较好的解。

时间复杂度：$O(nk + count \times k)$

空间复杂度：$O(nk)$

§72 PARSIN

§72.1 题目描述

给出M, N, X，求$k_1, k_2 \ldots, k_M$取遍满足$k_i \geq 0, \sum_{i=1}^{M} k_i = N$的所有向量$k$时，$\prod_{i=1}^{M} \sin(k_i X)$的和。

要求绝对误差或相对误差不超过0.1。

§72.2 数据范围

数据组数不超过10组，$1 \le M \le 30, 1 \le N \le 10^9, 0 \le X \le 6.28 < 2\pi$，答案不超过$10^{300}$。

§72.3 解题报告

令X固定，$f_{m,n}$为$M = m, N = n$时的答案，则

$$f_{m,n} = \sum_{i=0}^{n} f_{m-1,n-i} \sin(iX)$$

再引入

$$g_{m,n} = \sum_{i=0}^{n} f_{m-1,n-i} \cos(iX)$$

其中$g_{1,1} = \cos(X)$。

那么根据$\sin(a+b) = \sin(a)\cos(b) + \cos(a)\sin(b)$与$\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$，可得

$$\begin{cases} f_{m,n} = & \cos(X) f_{m,n-1} + \sin(X) g_{m,n-1} \\ g_{m,n} = & \cos(X) g_{m,n-1} - \sin(X) f_{m,n-1} + f_{m-1,n} \end{cases}$$

把$i + j = x$的$f_{i,j}$、$g_{i,j}$写在一排可以得到一个长度为$2M$的行向量V_x，可以找出矩阵M使得$V_x = MV_{x-1}$。直接上矩阵快速幂算法即可求得答案。

时间复杂度：$O(TM^3 \log N)$

空间复杂度：$O(M^2)$

§73 SHORTCIR

§73.1 题目描述

给出一个包含变量、and、or、not的布尔表达式，每个变量是独立的随机布尔变量，已知每个变量为true的概率。求最少情况下期望计算多少个变量的值可以确定整个表达式的值。

我们知道对于a and b类型的表达式，如果先计算a并知道了a = false，那么没有必要计算b。对于or表达式来说也是一样的。

优先级：not>and>or。

§73.2 数据范围

数据组数不超过50。表达式长度不超过30000。变量数不超过1000，变量名长度不超过5。表达式要么是合取范式(一些形如$((\neg)A_1 \vee (\neg)A_2 \vee \cdots \vee (\neg)A_n)$的式子and起来)，要么是析取范式(一些形如$((\neg)A_1 \wedge (\neg)A_2 \wedge \cdots \wedge (\neg)A_n)$的式子or起来)。

§73.3 解题报告

首先可以建立表达式树。可以用栈在线性时间内解决。

表达式树的叶子节点代表一个变量或它的取反。一个节点拥有以下属性：

- *not*：该节点是否被取反；
- *type*：该节点是其子树的and还是or；
- p：该节点答案为true的概率；
- *ans*：要计算出该节点的答案，期望计算多少个变量。

对于非叶子节点，很容易得到前3个属性；如果其$type$ = and，那么按照$\frac{ans}{1-p}$递增的顺序访问子树；否则按照$\frac{ans}{p}$递增的顺序访问。即可算出ans的值。这里用到了排序所以复杂度带log。

设L为所有表达式的长度和，V为变量数。

时间复杂度：$O(L\log V)$

空间复杂度：$O(L+V)$

§74 challenge SIMGRAPH

§74.1 题目描述

给出两个大小为n的无向图$G_1=(\{1\dots n\},E_1),G_2=(\{1\dots n\},E_2)$，求两个排列$P,Q$，使下式尽量大

$$\sum_{i=1}^{n}\sum_{j=1}^{n}[(P_i,P_j)\in E_1][(Q_i,Q_j)\in E_2]$$

§74.2 数据范围

$30\le N\le 75$，数据按某种方式随机生成。

§74.3 解题报告

固定其中一个排列为单位排列，用模拟退火算法调整另一个排列。产生相邻解的操作是交换排列中的两个元素。我们可以在$O(n)$的时间内得到新的解的价值。

时间复杂度：$O(n^2+count\times n)$

空间复杂度：$O(n^2)$

§75 challenge FAULT

§75.1 题目描述

给出$S\times N$的01矩阵A，从中选出尽量多的行，使得它们的秩不是N。保证$\text{rank}(A)=N$。

§75.2 数据范围

$50\le N\le 200, 2N\le S\le 2000$。数据按某种方式随机生成。

§75.3 解题报告

我用了两个程序。

程序1：随机选出一些列$i_1,i_2,\dots,i_m$，取出所有$A_{i_1}\text{ xor }A_{i_2}\text{ xor }\dots\text{ xor }A_{i_m}=0$的行。由于矩阵的列秩不是$N$所以行秩也不是$N$。多次随机取最优。我的程序选的是$m\le 3$的随机列集合。

程序2：由于第一个程序在某些测试点上效果较好而在另外测试点上效果一般，所以我随机选出这些行的一个线性基，将每一行用线性基表示出来，得到一个新的矩阵，运行程序1。

注意实现的时候压位。这样在tsinsen上可以获得101分。

时间复杂度：$O(ns+count\times\frac{sn^2}{32})$

空间复杂度：$O(\frac{sn^2}{32})$

§76 FLYDIST

§76.1 题目描述

给出一个N个点M条边的带权无向简单图，边权都是≤ 20的正整数。你可以进行一些操作，每次将一条边的权值加上$d(d\in\mathbb{Q})$，付出$|d|$的代价。你要使下列条件成立：对于任意两点(x,y)，如果它们之间有一条长度为w的边，那么它们之间的最短路长度为w。并且，边的权值始终>0。求付出代价的最小值，以分数形式输出。

§76.2 数据范围

$1 \le N \le 10, 1 \le M \le 45$。

§76.3 解题报告

对于一条边$e = (x, y, w)$，我们设给它加上了d_e^+的权值，减去了d_e^-的权值。设$g_{i,j}$为最后的图中i, j之间的最短路，如果图中i, j有边e那么$g_{i,j} = w(e) + d_e^+ - d_e^-$，否则$g_{i,j}$是一个新的变量。我们可以构建如下线性规划模型：

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} d_e^+ + d_e^- \\ \text{s.t.} \quad \forall i, j, k \in V, \quad & g_{i,k} \le g_{i,j} + g_{j,k} \\ \forall e \in E, \quad & d_e^+ \ge 0, d_e^- \ge 0 \end{aligned}$$

可以证明把边权变得< 1一定是不优的，所以不必加入$w(e) + d_e^+ - d_e^- > 0$这一条件。

用单纯形算法得到一个实数解，枚举分母即可得到分数形式。

记$M = O(N^2)$。变量有$O(N^2)$个，方程有$O(N^3)$个。

时间复杂度：$O(\text{simplex}(N^2, N^3))$

空间复杂度：$O(N^5)$

§77 challenge SPELL

§77.1 题目描述

Fedya和Alex是OIer。他们住在不同的城市里，但他们经常在网上讨论有趣的数据结构题、比赛、萌妹子、新番和其他一些奇奇怪怪的事情。

这个时候问题就来了，Fedya是一个非常急性子的人，每次刚输入完信息就会发送出去。尽管这没什么大问题，但跟我们每个人一样，Fedya有时候也会因为手抖而打错单词。每次看到有错别字的信息，有强迫症的Alex就会浑身难受。

确切地说，Fedya可能会犯下列错误：

- 1.交换了两个字母
- 2.漏掉了一个字母
- 3.多打了一个字母
- 4.打错了一个字母

Fedya毕竟是一个聪明的孩子，他不会在一个单词里犯一个以上的错误。

Fedya并不是不会拼这些词，所以出现错误的单词不会超过总单词数的4%。

Alex是一个好人，为了使自己不太难受，他决定自己写一个程序来改正Fedya的信息里的错误。但这对他而言并不是一个简单的工作，所以请你帮帮他。

正式地说，输入一段文本。它原本是一段语法和拼写正确的英文文本，但在某些地方发生了上述的错误。你需要输出修改后的文本，使得出现的错误尽可能少。为了使问题变得更简单，我们将给你提供一个字典。文中绝大多数单词都会出现在字典中，但不保证全部出现。

§77.2 数据范围

字典是给定的，单词数$D = 349900$，单词长度不超过$l = 12$，总大小约为$d = 3.05\text{MB}$。文本长度不超过$L = 10\text{MB}$。

§77.3 解题报告

对于文本的每一个单词，如果它直接在字典中出现，那么不需要改动；否则，枚举犯错误之前可能的单词(一共$O(\sigma l)$种)，如果出现在字典中就输出该单词。如果没有出现在字典中，则输出原单词。

这样能够做到tsinsen上112分。但是由于一些很短的常用单词在字典中没有出现(比如the,is)，所以如果单词长度过小(<4)就可以直接输出该单词。这样在tsinsen上可以拿到149分。

时间复杂度：$O(d+Ll\sigma)$

空间复杂度：$O(d+L)$

§78 LUCKYDAY

§78.1 题目描述

给出S_1,S_2,X,Y,Z,P,C,Q，定义数列S满足$S_i=(XS_{i-1}+YS_{i-2}+Z) \bmod p(i>2)$。有$Q$个询问，第$j$个询问求$L_j\le i\le R_j$的$i$中$S_i=C$的个数。

§78.2 数据范围

数据组数不超过2，$2\le P\le 10007$，P是质数，$1\le Q\le 20000$，$1\le L_i\le R_i\le 10^{18}$。

§78.3 解题报告

考虑A,B中有一个是0的情况。可以证明该数列的循环节是$O(P)$的，暴力即可。

否则该数列的循环节是$O(P^2)$的。

可以构造矩阵M使得$(S_{i+1},S_i,1)=M(S_i,S_{i-1},1)$。由$A,B\ne 0,P$是质数得知由$(S_{i+1},S_i)$可以推出$(S_i,S_{i-1})$，故$(S_1,S_2)$一定在循环节中。

令$r=O(\sqrt{P})$。

考虑$S_i=C$，令$i-1=pr-q$，其中$0\le q<r$。那么$(S_{i+1},S_i,1)=(S_2,S_1,1)M^{i-1}$，故$(S_{i+1},S_i,1)M^q=(S_2,S_1,1)(M^r)^p$。于是我们考虑BSGS算法。

枚举$q=O(\sqrt{P})$和$S_{i+1}=O(P)$，将向量$(S_{i+1},C,1)M^q$存入哈希表。

枚举$p=O(P\sqrt{P})$，询问$(S_2,S_1,1)(M^r)^p$的出现个数。

令$S_{i+1}=S_2,S_i=S_1$还可以求出循环节len。

按照上面方法可以对每个p求出$[pr-r+2,pr+1]$中的C的个数，再预处理出$S_1,S_2,S_{r+1},S_{r+2},S_{2r+1},S_{2r+2},\ldots$共$O(P\sqrt{P})$个元素，即可在$O(\sqrt{P})$的时间内回答询问。

时间复杂度：$O(T(P+Q)\sqrt{P})$

空间复杂度：$O(P\sqrt{P})$

§79 BB

§79.1 题目描述

大厨在公路旁建了一个餐馆。但根本没有人来用餐，以至于经营这个餐馆让大厨大亏一笔。大厨觉得这一定是广告没有做好，于是他决定在公路的一些广告牌上放餐馆的广告。

在公路上有N个广告牌，大厨会选择一部分广告牌放广告。为了加深路人的影响，大厨希望任意连续的M个广告牌中至少有K个餐馆广告。由于之前一直在亏钱，大厨希望在满足条件的同时使用最少的广告牌。现在大厨想知道有多少种使用广告牌的方案。模$mod=10^9+7$。

§79.2 数据范围

数据组数不超过300，$1\le K\le M\le 500$，$M\le N\le 10^9$。

§79.3 解题报告

考虑$M\mid N$的情况，使用的广告牌数量一定是$\frac{NK}{M}$。将广告牌每M个分为一组，令$f_{i,j}$表示第i组中第j个被使用的广告牌在第i组中的编号。我们发现一组解是合法的，当且仅当$1\le f_{i,j}\le M$，且$f_{i,j}\le f_{i+1,j}(i<\frac{N}{M})$，$f_{i,j}<f_{i,j+1}(j<K)$。

我们令$n=\frac{N}{M}$。这是一个半标准杨氏矩阵，其方案数为：

$$ans=\prod_{i=1}^{n}\prod_{j=1}^{K}\frac{M-j+i}{n-i+K-j+1}$$

令$A_i=\prod_{j=1}^{K}(i+j)$，那么

$$ans=\prod_{i=1}^{n}\frac{A_{M-K-1+i}}{A_{i-1}}=\frac{\prod_{i=n}^{M-K-1+n}A_i}{\prod_{i=0}^{M-K-1}A_i}$$

计算这个式子的复杂度是$O(MK\log mod)$的。

当$M\nmid N$的时候，如果$N \bmod M\le M-K$，那么每一组前$N \bmod M$个值都是0，这时我们只要令$M\leftarrow M-N \bmod M$即可；否则每一组后$M-N \bmod M$个值都是1，需要进行$n\leftarrow n+1,K\leftarrow K-M+N \bmod M,M\leftarrow N \bmod M$。

§80 challenge BICKER

§80.1 题目描述

无向完全图$G=(V,E)$的每条边有两个权值d和q，求一个割使得$\frac{\sum q}{\sum d}$最小。不允许$\sum d=0$。

§80.2 数据范围

数据组数不超过30，$d\ne 0$的边数D、$q\ne 0$的边数Q不超过10^4，$2\le n\le 500$，$1\le d_i,q_i\le 10^4$。

§80.3 解题报告

可以用割的一边的集合(S)来表示状态，每次转移就是把一个元素扔进S或从S中踢出。跑模拟退火算法即可。

不过这题如果随机生成初始解的话效果很不好。我们随机两个点作为源和汇，以q为权值跑最小割，跑多次得到的方案中最优的作为初始解即可。

时间复杂度：$O(k\times \text{networkflow}(n,Q)+count\times n)$

空间复杂度：$O(D+Q)$

§81 challenge CKROACH

§81.1 题目描述

所有的厨师都有一个最大的敌人，那就是G。有一天，厨师发现餐厅里有n只G，他非常害怕，打算用杀虫剂杀死所有的G。现在有m种可用的杀虫剂，第j种杀虫剂的价格是C_j。不幸的是，G可能有抗药性，也就是说G可能在厨师用了杀虫剂后还活着。但是通过统计我们知道，如果厨师使用第j种杀虫剂，第i只G将会有$P_{i,j}$%的概率死亡(这些概率是独立的)。

厨师想买一些杀虫剂，然后一种接一种地使用(每种杀虫剂只能使用一次)。他希望在他使用完杀虫剂之后，所有的G都有至少90%的概率死亡。当然，厨师想要花尽可能少的钱，所以他希望你帮助他选择一种更优的方案。

§81.2 数据范围

数据组数不超过10，$50\le n\le 200,50\le m\le 200$，$1\le C_j\le 10^6,0\le P_{i,j}\le 90$。

§81.3 解题报告

设选的杀虫剂编号为$i_1,i_2,\dots,i_k$。则需要满足$\forall 1\le x\le n,\prod_{j=1}^{k}(1-\frac{P_{x,i_j}}{100})\le 10\%$。

令$Q_{i,j}=-\log_{10}(1-\frac{P_{x,i_j}}{100})$，那么$\forall 1\le x\le n,\sum_{j=1}^{k}Q_{i,j}\ge 1$。

首先按顺序遍历杀虫剂，如果它能够增加某个死亡概率$<90\%$的G的死亡概率，那么使用它。可以得到一个初始解。接下来每次选择一个杀虫剂i，如果它被使用了且删掉它之后的解不合法，那么就

重新选择。如果i没有被使用那么把它加入当前的解中，否则把它从当前的解中踢出去。用这种方法来产生相邻解。跑模拟退火算法即可。

时间复杂度：$O(mn + count \times n)$

空间复杂度：$O(mn)$

§82 challenge SEAND2

§82.1 题目描述

Sereja有一个整数A，在它的十进制表示中不含零。同时，他有N个整数，$B_1, B_2, \ldots, B_N$。

首先，我们定义关于数字A的一个函数f：

$$f(A) = \sum_{i=1}^{N} (A \bmod B_i)$$

现在Sereja想要对A中的数字进行重排，来最小化$f(A)$的值。请你帮他找到尽可能优的解。

§82.2 数据范围

数据组数不超过100，$1 \le A \le 10^{1000}, N = 100, 1 \le B_i \le 10^6$，$A$是随机生成的。

§82.3 解题报告

直接进行退火，以A本身作为状态。每次交换两个数位的值，可以在$O(N)$时间内计算出新解的f值。配合上卡时就有很好的效果了。

时间复杂度：$O(|A|N + count \times N)$

空间复杂度：$O(|A|N)$

§83 challenge MAXRECT

§83.1 题目描述

给出一个$N \times M$的矩阵，要求选出一些行和一些列，最大化这些行列相交处元素的和。

§83.2 数据范围

$200 \le N, M \le 300$，元素的范围是$[-10^9, 10^9]$。

§83.3 解题报告

以每一行或每一列是否被选为状态进行退火。产生新解的规则：每一次随机一行或随机一列，如果它本来在状态内就把它踢出；否则把它加入状态。可以在$O(N)$或$O(M)$的时间内得到新解的价值。

初始解的话。。随便乱选就好了，我的方法是遍历所有行，找出这一行中的所有正数加起来，即

$$\max_{i=1}^{N} \sum_{j=1}^{M} \max(A_{i,j}, 0)$$

时间复杂度：$O(NM + count \times (N + M))$

空间复杂度：$O(NM)$

§84 GERALD08

§84.1 题目描述

一棵N个节点的有根树T，每条边要么是黑色要么是白色。两个玩家C和K轮流操作，要操作的玩家选择一条还在树上的边，然后删掉这条边，之后不与根连通的部分也会被删掉。C只能删黑边，K只能删白边。

问在C/K是先手的情况下谁有必胜策略。

§84.2 数据范围

数据组数不超过1000，所有数据中N的和不超过10^5。

§84.3 解题报告

对于这种不平等博弈问题我们可以对每个局面求出其值(一个Surreal Number，在本题中一定是分母是2的幂的实数)。一个局面的值是一个surreal number $S = \{L|R\}$，其中L是该局面删除一条黑边之后的新局面的值，R是该局面删除一条白边之后的新局面的值。如果$S > 0$，那么C必胜；如果$S < 0$，那么K必胜；否则后手必胜。

现在我们的任务就是求出输入局面的权值。

如果根的度数不为1，假设其有k个子树，$S_1, S_2, \ldots, S_k$分别表示第i个子树**以及其与根相连的那条边**。那么这棵树的值是所有子树的值的和。

如果根的度数为1，其子树的值为x，与根相连的边是黑的。我们找出最小的正整数p使得$x+p > 1$，那么当前局面的权值为$\frac{x+p}{2^{p-1}}$；如果与根相连的边是白的，我们找出最小的正整数p使得$x - p < -1$，当前局面权值为$\frac{x-p}{2^{p-1}}$。

考虑用高精度数维护每个节点的子树的值。具体地说，对每个值A我们维护一个数集S(可以用STL的 set <int>实现)和两个整数I, w，表示

$$A = I + \sum_{j \in S} 2^{j+w}$$

并且保证$\forall j \in S, j + w < 0$。可以证明，对于一个大小为$s$的子树，$|I| = O(s)$。

我们需要支持的操作有：

- 加上一个int。直接把该int与I相加即可。复杂度$O(1)$。
- 除以$2^p (p > 0)$。把w减去p。然后把I的后p位中所有的1放入集合s中。复杂度$O(p \log N)$。不过实践中p好像不大，我也没有找到p的一个上界，只知道$I > 0$时$p = O(\log N)$，那就算作$p = O(\log N)$吧。
- 加上另一个数(S', I', w')。$I \leftarrow I + I'$，然后对于S'中的每个数j，把$j + w' - w$放入集合S。如果$j + w' - w$本来就在S中，那么我们找出最小的k使得$k \sim j + w' - w$都在S中，把它们都删掉。如果$k - 1 + w = 0$那么$I \leftarrow I + 1$；否则将$k - 1$插入S。复杂度的话，第一部分是遍历S'的复杂度，就是$O(|S'| \log N)$，如果使用启发式合并那么总复杂度是$O(N \log^2 N)$；第二部分是删除的复杂度，加起来不会超过$O(N \log^2 N)$。

对每一个节点维护一个高精度数表示它的权值，从下往上递推求出根的权值，再跟0比较即可。

时间复杂度：$O(N \log^2 N)$

空间复杂度：$O(N \log N)$

§85 LECOINS

§85.1 题目描述

有N种类型的硬币，第i种的价值是V_i，颜色是C_i，每种硬币有无限个。有Q个询问，每次问用一些硬币组成币值S，至多可以用多少种颜色。

§85.2 数据范围

$1 \le N \le 30, 1 \le V_i \le 2 \times 10^5, 1 \le C_i \le 10^9, 1 \le Q \le 2 \times 10^5, 1 \le S \le 10^{18}$。

§85.3 解题报告

令$v = V_1$。

用$f_{i,j}$表示使用i种颜色支付的币值模v为j，那么这个币值至少是多少。

把硬币按颜色排列，即，相同颜色的硬币靠在一起。依次处理每一枚硬币。

如果这枚硬币与上一枚硬币颜色不同，那么我们遍历该颜色的所有硬币V_x，对所有的(i,j)进行松弛：$f_{i,j} \leftarrow \min(f_{i,j}, f_{i-1,(j-V_x) \bmod v} + V_x)$。

不论硬币颜色如何，我们需要更新f数组。设该硬币的价值为v'。对于每一个i，我们建立一张图，包含编号为$0 \to v-1$的点和一个源点s。对于每个j，s向j连权值为$f_{i,j}$的边；j向$(j+v') \bmod v$连权值为v'的边。s到每个点的最短路就是新的f_i数组。

可以这样求最短路：去掉与s相连的边之后图形成了若干个环，对于每个环找出与s连边权值最小的点k，那么$(k-v') \bmod v$向k连的边没有用。从k开始递推即可求出最短路。

预处理f数组的复杂度是$O(N^2M)$的。

考虑询问S。我们保留两个版本的f数组：f_1(没有考虑与硬币1颜色相同的所有硬币)和f_2(强制要求选与硬币1颜色相同的硬币)。首先拿-1更新答案；如果$f_{1,i,S \bmod v} = S$，拿i更新答案；如果$f_{1,i,S \bmod v} < S$，拿$i+1$更新答案；如果$f_{2,i,S \bmod v} \le S$，拿i更新答案。答案是所有更新中的最大值。

时间复杂度：$O(N^2M + NQ)$

空间复杂度：$O(NM)$

§86 EVILBOOK

§86.1 题目描述

厨师Ciel从《邪恶之书》中获得了巨大的烹饪力量。

然而，Ciel必须贡献666点魔法力量给邪恶之书。

除了Ciel，世界上还有N个厨师。

第i位厨师有C_i点烹饪力量和M_i点魔法力量。

Ciel可以通过在一次烹饪战斗中打败第i位厨师获得M_i点魔法力量，同时需要付出C_i点努力。

在这场战斗之后，第i位厨师的魔法力量M_i变为0。

如果她给邪恶之书X的魔法力量，邪恶之书可以帮助她。

也就是说，通过消耗X点魔法力量，她可以选择某个i并把第i位厨师的C_i和M_i都乘以$\frac{1}{3}$。

注意，只有当她拥有至少X点魔法力量时才能请求邪恶之书的帮助。

初始时，Ciel没有魔法力量。

至少总共需要多少努力才能收集到至少666点魔法力量?

§86.2 数据范围

数据组数不超过5，$1 \le N \le 10$，$10 \le X < 666$，$0 \le C_i, M_i \le 10^9$。

§86.3 解题报告

考虑搜索。每次选出一个$M_i \ne 0$的厨师i，枚举对它使用的帮助次数k_i(即，该厨师的$C_i \leftarrow \frac{C_i}{3^{k_i}}$，$M_i \leftarrow \frac{M_i}{3^{k_i}}$)，然后打败该厨师，继续搜索。

这样显然是过不了的，我们考虑剪枝。

- 首先，如果$\frac{M_i}{3^{k_i}} \le k_i X$，那么打败该厨师显然是没有意义的。
- 对于一个厨师，如果还能使用帮助，且使用了帮助之后再打败他(仍然)可以集齐≥ 666魔法力量，那么我们**没有理由不使用帮助**。

加上上述两个剪枝之后就可以搜索出解了。

时间复杂度：$O(玄学)$

空间复杂度：$O(N)$

§87 CHEFBOOK

§87.1 题目描述

给出N和M个整数五元组$(x_i, y_i, L_i, S_i, T_i)$，求两个长度为$N$的整数数组$P, Q$，满足对于每个五元组$(x, y, L, S, T)$，有

$$S \le L + P_x - Q_y \le T$$

并且P, Q中的元素都必须在$[0, 10^6]$之间。

对于每个五元组(x, y, L, S, T)，它对答案的贡献为$L + P_x - Q_y$。最大化这个答案。

要求输出方案。可能无解。

§87.2 数据范围

数据组数不超过10，$1 \le N \le 100$，$1 \le M \le N^2$，$L \in [-600, 600]$，$S, T \in [-1000, 1000]$。

§87.3 解题报告

我们列出线性规划

$$\begin{aligned}
\text{maximize} \quad & \sum_{i=1}^{M} P_{x_i} - Q_{y_i} \\
\text{s.t} \quad \forall 1 \le i \le M, \quad & P_{x_i} - Q_{y_i} \le T_i - L_i && (3) \\
& Q_{y_i} - P_{x_i} \le L_i - S_i && (4) \\
\forall 1 \le i \le N, \quad & P_i, Q_i \ge 0
\end{aligned}$$

考虑其对偶问题，令(3)对应的变量为$X_{1\ldots M}$，(4)对应的变量为$Y_{1\ldots M}$，那么：

$$\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{M} (T_i - L_i)X_i + (L_i - S_i)Y_i \\
\text{s.t} \quad \forall 1 \le j \le N, \quad & \sum_{i=1}^{M} [x_i = j](X_i - Y_i) \ge \sum_{i=1}^{M} [x_i = j] && (5) \\
& \sum_{i=1}^{M} [y_i = j](Y_i - X_i) \ge -\sum_{i=1}^{M} [y_i = j] && (6) \\
\forall 1 \le i \le M, \quad & X_i, Y_i \ge 0
\end{aligned}$$

所有(5),(6)式相加，出现了$0 \ge 0$的场景，所以等号是可以取到的。

对于每一个变量X_i或Y_i，它只出现在(5)的第x_i个方程中和(6)的第y_i个方程中，且系数一次为1一次为-1。建立一个$2N$个点的费用流图，点编号$A_1, \ldots, A_N, B_1, \ldots, B_N$，代表对偶问题中的方程。把对偶问题中的变量当成边，容量为$+\infty$，值当成流量，minimize中的系数当成费用。对于每个点我们列出"入流量$-$出流量$= 0$"。对于一个$1 \le i \le M$，$A_{x_i} \to B_{y_i}$连一条边，费用为$L_i - S_i$，代表Y_i；$B_{y_i} \to A_{x_i}$连一条边，费用为$T_i - L_i$，代表X_i。建立源s和汇t以补充(5),(6)右边的值。考虑一个式子i右边系数为f，如果$f > 0$那么i向t连费用为0容量为f的边；否则s向i连容量为$-f$费用为0的边。跑一遍最小费用最大流可以得到对偶问题的最优解。使用互补松弛定理求出原问题的最优解：

定理7. *如果对偶问题中的某一变量x在最优解中不为0，则原问题中x对应的约束条件取严格等号。*

原问题中每个限制形如$a - b \le c$，如果需要取严格等号那么增加一个限制$b - a \le -c$。

构建差分约束系统：以原问题的变量为点，设立一个源点，变量的值等于源点到它的距离。如果$i - j \le k$，那么$j \to i$连边，权值为k。然后跑一遍最短路即可。

此时跑出来的解可能不在$[0, 10^6]$的范围内，同时加上一个数(它们最小值的相反数)是没有问题的。

无解的情况来自于费用流图的负权环，此时对偶问题的值域无穷，原问题即无解。

时间复杂度：$O(\text{costflow}(N, M) + \text{shortestpath}(N, M))$

空间复杂度：$O(M)$

§88 challenge SIMNIM

§88.1 题目描述

给出N个数，值域$[1, 2^M - 1]$，已知它们的异或和为0。请将它们分成若干集合，使得每个集合的异或和都为0。最大化分出来的集合数。

§88.2 数据范围

10组数据，$10 \le N \le 1000, 5 \le M \le 60$，数据随机生成。

§88.3 解题报告

进行若干轮这样的操作：选出这些数的一个线性基，将其他数用线性基表示出来。对于每个基外的数a_i，我们知道它用了多少个基中的数才能表示出来，记为f_i。在构建基的时候可以顺带记录一些信息来得到a_i是被哪些数表示出来的。然后把这些数(一共f_i+1个)分到一个集合内。重复这样的过程，直到所有数都在某个集合内为止。

时间复杂度：$O(N^2M)$

空间复杂度：$O(N)$

§89 LMATRIX3

§89.1 题目描述

有一个数字P和一个长度为N的数组，记数组的第i个元素为A_i。每次操作可以选择三个整数s,t和k，满足：$1 \le s \le t \le N$且$0 \le k \le 10000$。然后，将对所有满足$s \le i \le t$的A_i进行变换$A_i \leftarrow (k+A_i) \bmod P$。求将$A$数组变成全0数组的最少次数。

N很大，所以读入的数组都是由M个长这样的小数组B_i拼起来的：

$$B_{i,1} = F_i, \forall 1 < j \le L_i, B_{i,j} = (B_{i,j-1}C_i + D_i) \bmod P$$

§89.2 数据范围

数据组数不超过100，$1 \le M \le 100$，$1 \le P \le 10$，$1 \le L_i \le 10^{15}$，$0 \le F_i, C_i, D_i < P$。

§89.3 解题报告

首先将数组差分，得到$N+1$个元素$a_1, a_2, \ldots, a_N, a_{N+1}$，其中$a_i = A_i - A_{i-1}$，$A_0 = A_{N+1} = 0$。一次操作相当于取出一个$a_i$和一个$a_j(i \ne j)$，将$a_i \leftarrow (a_i + k) \bmod P$，$a_j \leftarrow (a_j - k) \bmod P$。明显我们只需要弄到数组$c_{0 \ldots P-1}$，$c_i$表示$a_x = i$的$x$的个数。

注意到B数组的循环节都不超过P，所以搞到循环节之后可以很容易弄到数组c。

考虑把一次操作的i, j之间连一条无向边，可以证明一个连通块内数的原来权值之和为0。最优解中不会有环，因为对于一个大小为n的连通块显然可以用$n-1$次操作把它们都变成0。所以操作个数就是$N-$连通块个数。于是问题变成：把a序列分成尽量多个小集合，使每个集合中数的和都是0。

接下来我们选出尽量多的$a_i + a_j = P$的数对，一次操作可以把它们都变成0。换句话说：

$$(c_i, c_j) \leftarrow (c_i - \min(c_i, c_j), c_j - \min(c_i, c_j))(i \ne j, i + j = P)$$

$$c_i \leftarrow c_i \text{ and } 1(2i = P)$$

这显然是最优的。

然后可以注意到数的个数不会很多(不超过5个)，小集合的特征是“和为0且不存在和为0的非空真子集”。那么可能的小集合个数也不会很多。一共不超过$S = 30$个。可以求出所有小集合。对每一个小集合设一个变量表示出现次数，那么可以列出一些方程，变成一个整数线性规划问题。

对于几乎所有情况，答案等于线性规划问题的解取整。直接跑单纯形就好。

因为数据达到了10^{17}级别，所以我们可以用`decimal128`来进行较高精度的浮点数操作。

需要`#include <decimal/decimal>`和`using namespace decimal;`

时间复杂度：$O(MP+\text{simplex}(S,P))$

空间复杂度：$O(SP)$

§90 challenge WORDNINJ

§90.1 题目描述

注意！这道题很长，你可能需要花上一些时间来看题。

我们来玩WordNinjas吧！你需要使用你的ninja得到的方块上的字母来组成单词。有些特殊方块拥有额外加分。

在这里我们考虑这个游戏的简化版本：所有的方块排成一行并且我们事先知道他们的得分。这些方块被分成几个大块，每一步你可以跳过当前的大块或者在这个大块中取走一个方块。你能从输入中获得一个大块的序列，每个大块是一个字符串，其中每个字母都代表一个方块。每个方块可以是一个字母方块或者一个特殊方块。字母方块将使用对应的大写字母表示。特殊方块一共有5种，其中第一种是空白方块，它的作用与字母方块相似，但是你可以任意指定它上面的字母（具体介绍见下文）。

这个游戏的目标是获得尽可能多的分数。分数的主要来源是组词。为了使你能够组词，游戏提供了一个长度为7的数组，标号为$1\sim7$，里面可以填入字母方块和空白方块。数组中的每个位置都只能放入一个方块。假如在某一次你选取了一个字母方块或空白方块后数组已经放满了7个方块，那么这个方块会被丢弃。否则这个方块会占据数组中第一个空的位置。所以每时每刻数组中的方块都是开始的几个连续方块。你随时都可以删除数组中的某一个方块，剩下的方块将会向前移动直到满足要求。

最后，你可以任意排列数组中的方块，空白方块可以视为任意一个字母。根据你拼出的单词你将会得到一些分数（见下文）。之后除了最后一个方块以外的方块将会被丢弃，最后一个方块将被放到数组的第一个位置。

关于5种特殊方块：

- 空白方块：使用"?"表示。和之前提到的一样，这个方块可以当成一个字母方块使用。取走它你可以获得5分。在计算单词得分时视为0分。
- 双倍字母得分方块：使用"1" "7"表示。使某一个位置$(1\sim7)$的字母得分$\times2$。
- 三倍字母得分方块：使用"a" "g"表示。使某一个位置$(1\sim7)$的字母得分$\times3$。
- 双倍单词得分方块：使用"+"表示。使所有位置的字母得分$\times2$。
- 三倍单词得分方块：使用"*"表示。使所有位置的字母得分$\times3$。

取走以上四种特殊方块将会得到2分。注意：数组的每个位置都记录了一个倍数信息，初始为1。假如你取走了一个双倍字母得分方块，那么对应的位置的倍数信息将会变为2，而不管他原来是几。即假如它原来是3，你取走了一个双倍字母得分方块，那么你的得分将会下降。同理，整个数组也会记录一个倍数信息。当你组出一个词后，所有的倍数信息将会变为1。

以下是每个字母的得分：

1分 A, E, I, L, N, O, R, S, T, U
2分 D, G
3分 B, C, M, P
4分 F, H, V, W, Y
5分 K
8分 J, X
10分 Q, Z

设L为你组出的词长度$(2\le L\le7)$，C_i为第i个字母的分数（注意空白方块的分数为0），M_i代表每

个位置的倍数信息，M代表整个数组的倍数信息，你的得分为：

$$M\sum_{i=1}^{7} M_iC_i + 3\max(L-3,0) + 30\max(L-6,0)$$

即长度为4,5,6,7的词分别能得到3,6,9,42的加分。

你的总分为组出的单词的得分与取走特殊方块的得分之和。你的任务是最大化你的得分。

§90.2 数据范围

所有数据中的字典都是相同的，它包含$D=74414$个长度为$2\sim 7$的单词。每个测试点的每个大块中的小块数量之和是$L=10000$。其中9000个是字母或空白方块。其中每100个包含2个空白方块和98个字母方块。字母方块的比例是事先确定的。剩下的方块都是特殊方块，总共有30个三倍单词得分方块，70个双倍单词得分方块，300个三倍字母得分方块和600个双倍字母得分方块。每个翻倍字母得分方块对应的数组位置都是随机生成的。每个大块的大小是在$1\sim 10$之间随机生成。

§90.3 解题报告

由于长度为7的单词加分特别高，所以只考虑它们。并且只考虑得分最高的$X=1100$个单词。

考虑一些块以及一个单词，可以用二分图匹配的模型判断这些块是否可以得到该单词。左边的点表示块，右边的点表示单词中的字符，一个块中有一个字符那么连边。如果最大匹配为单词长度那么是可行的。

当多加一个块的时候，直接从这个块开始跑增广路就好。

如果发现了可以匹配的单词那么匹配。中间一些块如果没有用上，且有加分方块那么直接选。

时间复杂度：$O(XL+D\log D)$

空间复杂度：$O(X+L+D)$

§91 challenge MANYLEFT

§91.1 题目描述

给出一个$N\times N$的01矩阵A。每一次可以选择两个位置(x_1,y_1)与(x_2,y_2)，要求以下条件同时满足：

- $x_1=x_2$且$|y_1-y_2|=2$，或者$y_1=y_2$且$|x_1-x_2|=2$;
- $A_{x_1,y_1}=1$，$A_{x_2,y_2}=0$;
- 令$x_m=\frac{x_1+x_2}{2}$，$y_m=\frac{y_1+y_2}{2}$，则$A_{x_m,y_m}=1$。

然后将$A_{x_1,y_1},A_{x_m,y_m},A_{x_2,y_2}$都取反。

要求进行一系列操作到**不能再进行操作了为止**。进行的操作次数越小，分数越高。

§91.2 数据范围

$10\le N\le 30$，$P\in[0.5,0.95]$，每个$A_{i,j}$有P的概率是1。

§91.3 解题报告

我们发现以下贪心效果比较好：为每个局面设计一个估价函数，每次选择估价函数值最高的走法走。我设计的估价函数是相邻的1的数目($(a,b),(b,a)$算一次)乘2.5加上下一步可能的操作数。

在修改一个位置的值的时候，可以$O(1)$地算出新的估价函数值。

我们给贪心加入随机化：每一步有95%概率选择估计函数值最高的走法，5%概率随机选择一种走法。尝试多次取最优解，卡时。

时间复杂度：$O(N^2+count\times(N^2+ans))$

空间复杂度：$O(N^2)$

§92 MAXCIR

§92.1 题目描述

给出一个三角形ABC，以及N个操作。第i个操作有两个参数x_i, y_i，使用这个操作可以使得点A的x坐标增加x_i，并且y坐标增加y_i。

你可以使用最多K个操作，它们的影响叠加。同一个操作不能重复使用，ABC允许共线或重合。

最大化三角形ABC的周长。答案的**绝对误差**必须小于10^{-12}。

§92.2 数据范围

$K \le N \le 500$，$|A_x|, |A_y|, |B_x|, |B_y|, |C_x|, |C_y| \le 10^9, |x_i|, |y_i| \le 10^6$。

§92.3 解题报告

首先我们需要最大化的就是$|AB| + |AC|$。

然后，一定存在u, v，使得在最大化$uA_x + vA_y$的时候最大化了$|AB| + |AC|$。因为，考虑最优解对应的点A，记$l = |AB| + |AC|$，所有$|AB| + |AC| \le l$的点组成了一个椭圆，A在椭圆的边界上。令椭圆过A点的切线为$ux + vy = C$或$-ux - vy = C$即可。

假设我们确定了u, v，对每个操作i求出$ux_i + vy_i$，取最大的前K个中的所有正数对应的操作即可。我们称对于一个确定的(u, v)，它的**操作排列**是所有操作按照$ux_i + vy_i$排序后得到的排列，**正操作集合**是所有$ux_i + vy_i > 0$的操作组成的集合。

考虑操作排列和正操作集合都相等的一些(u, v)，它们确定了一个极角区间。可以发现，这些极角区间一共只有$O(N^2)$个，分界线分别是：

- 对于一个操作i，令$ux_i + vy_i = 0$得到的(u, v)；
- 对于两个操作i, j，令$ux_i + vy_i = ux_j + vy_j$得到的$(u, v)$。

处理出这$O(N^2)$个分界线，排序，然后从每个区间中选一个代表向量(例如两边界向量的和)，求出其操作排列，以及操作排列上x, y坐标的前缀和，二分最后一个正数操作，求出答案。

我们发现求一次操作排列是$O(N \log N)$的，太慢。考虑维护操作排列，从一个区间走到另一个区间的时候会跨过一个分界向量，如果它是i, j求出的向量，那么交换i, j在操作排列中的位置，再更新前缀和即可。有一个不好处理的边界情况就是一些分界向量同向共线。在这些分界向量之间不选出代表向量。当跨过其中一个向量时跨过所有向量。交换i, j的时候，拿后一个区间的代表向量(u, v)查一下，如果$ux_i + vy_i$与$ux_j + vy_j$满足交换条件则交换。记录所有交换过的值，按顺序访问它们，访问的时候更新前缀和。另外，如果全部向量都共线，需要特判。

最后是误差的问题，令$\sqrt{x} = I + D, I \in \mathbb{Z}, D \in [0, 1)$，则$x = I^2 + 2DI + D^2$，$D = \frac{x - I^2}{I + \sqrt{x}}$。在计算答案(用于比较)的时候需要用这样更精确的开方函数才能保证精度。

时间复杂度：$O(N^2 \log N)$

空间复杂度：$O(N^2)$

§93 challenge CHAORNOT

§93.1 题目描述

给出一个数集S，选出其尽量大的子集S_0，使得S_0中不存在三个不同的数i, j, k使得$i + k = 2j$。

§93.2 数据范围

L是$[10^4, 10^5]$中的随机整数，p是$[0.1, 0.9]$中的随机实数。$0 \ldots L - 1$有p的概率被加入S中。

§93.3 解题报告

考虑一个贪心：每次选一个数，如果把它加入之后S_0仍然满足条件，那么把它加入；否则不加入。由于S_0的大小实际上不大，所以这一步可以暴力。

用这个贪心产生一个初始解，然后进行退火等随机调整。每次随机选取一个数，如果它不在S_0中且把它加入S_0会破坏条件，那么重新随机；否则把它在S_0中的状态取反，即可得到一个新的解。

时间复杂度：$O(|S||S_0| + count \times |S_0|)$

空间复杂度：$O(\text{值域} + |S|)$

§94 challenge MAXDIFFW

§94.1 题目描述

这是一道交互题。

有一个N个点的有向完全图，边权$w(i,j)$是$1 \sim N(N-1)$的不同正整数，边权分布随机。有一个棋子在节点间移动。第一步，先手把棋子放在某个点u上并将其沿着一条边(u,v)移动到v，并获得$w(u,v)$的得分。然后两人轮流，每次把棋子移动到一个相邻节点，要求之前走过的边不能再走。一个人走过一条边，他就得到这条边的权值。直到不能走为止。要求你操控先手跟后手博弈，最大化你的权值减去对方的权值。

后手的策略有两种：

- MAX策略。每次选择可以走的边中权值最大的。
- RANDOM策略。每次随机选择一条可以走的边。

§94.2 数据范围

$N = 50$。

§94.3 解题报告

首先我们判断后手的策略。默认其为MAX，当他走了一次非最大边的时候，认为是RANDOM。

记录N个堆，每个堆按权值记录每个点的所有出边。下文中的Max函数可以$O(1)$求。

考虑后手策略为MAX的情况。

令$f_{i,j}$为你做先手从点j走，你走i次以内能够得到的最高分。因为f函数只是用来估计，所以不需要考虑重复经过一条边。

$$f_{0,j} = 0, f_{i,j} = \max_{(j,k)\in E} w(j,k) - w(k, Max(k)) + f_{i-1,Max(k)}$$

其中$Max(k)$为k出发的最大边指向的节点。

再令f_j为所有$f_{i,j}$中的最大值。

我们令$d = 20$，每次Move的时候算出新图的$f_1 \dots f_d$，复杂度$O(dN^2)$。

每次搜索你的决策：令上一条边为(u,v)，那么你枚举你的决策p，最大化

$$w(v,p) - w(p, Max(p)) + Kf_{Max(p)}$$

其中$K = 0.5$是我们确定的参数。这样可以在tsinsen上拿到55分(10个MAX点)。

再考虑RANDOM，照样求出$g_{i,j}$表示你做先手从点j走，你走i次以内的期望最高分。

$$g_{0,j} = 0, g_{i,j} = \max_{(j,k)\in E} w(j,k) + \frac{\sum_{(k,l)\in E} g_{i-1,l} - w(k,l)}{\sum_{(k,l)\in E} 1}$$

注意到上面那个dp还是可以$O(N^2)$转移的：有一重循环只与k有关，预处理即可。

同样，求出所有$g_1 \dots g_{d'}$，令g_j为所有$g_{i,j}$中最大值。因为g部分涉及实数，常数较大，可以令$d' = 15$。复杂度中$d' = O(d)$。

跟MAX一样，令上一条边为(u,v)，枚举决策p，最大化

$$w(v,p)+\frac{\sum_{(p,q)\in E}Kg_q-w(p,q)}{\sum_{(p,q)\in E}1}$$

这里我没有得到很好的运行结果，调出来最好的$K=1$。tsinsen上只拿了45分。

时间复杂度：$O(dN^2steps)$，其中$steps$为Move的调用次数。

空间复杂度：$O(N^2+dN)$

§95 COOLNUM

§95.1 题目描述

一个数N有K位，每个数位上的数字分别为$X_1,X_2,\ldots,X_K$，如果N存在一到三个数位上的数的和为s，且$(X_1+X_2+\cdots+X_K-s)^s$是N的倍数，那么N是cool number(CN)。

定义$LC(N)$和$RC(N)$分别是$\le N$的最大的CN和$> N$的最小的CN，多次询问给定N，求$LC(N)$和$RC(N)$。

§95.2 数据范围

数据组数不超过10^5，$1\le N\le 10^{1000}$，所有N的长度和不超过4×10^6。

§95.3 解题报告

存在两类CN：

第一种是存在不超过3个非0位的数。这样的数都是CN。要找出只考虑第一类CN的$LC(N)$，那么贪心地保留最前面三个非零数位即可。求$RC(N)$可以把最后一个非零数位+1。

第二种是除第一种以外的所有数。注意到$s\le 27$，即$10^K\mid(9K)^{27}\Rightarrow 10^K\le(9K)^{27}\Rightarrow K<77$。所以第二种CN是有限的，考虑预处理出所有第二种CN。枚举$sum=X_1+X_2+\cdots+X_K$，s，$(sum-s)^s$的所有约数，逐一判断。可以发现只有$M=32740$个第二种CN。

但是这样搜索会TLE。我们发现能够搜出答案的(sum,s)打一个表出来，主程序搜索的时候只搜索指定的(sum,s)。还有一个优化：一些(sum,s)只能搜出一个解，我们把这些解也打出来，主程序只搜索多于一个解的(sum,s)。

询问的时候，如果数字长度≤ 3，直接输出$N,N+1$。否则，首先答案赋值为第一种CN的答案；如果数字长度≤ 80，在第二种CN排序后的表中二分查找，更新答案。注意细节不要写挂了。

时间复杂度：$O(\sum(80\log M+\log N))$

空间复杂度：$O(M+\log N)$

§96 SHORT

§96.1 题目描述

给定两个数n,k，求数对(a,b)的个数，满足$n<a,b<k$，且$ab-n$可以被$(a-n)(b-n)$整除。

§96.2 数据范围

$0\le n\le 10^5,k\le 10^{18}$。

§96.3 解题报告

当$n=0$时，答案为$(k-1)^2$。需要高精度。下面假设$n>0$。

令$ab-n=p(a-n)(b-n)$。显然$p>1$。则$b=n+\frac{n(a-1)}{p(a-n)-a}$。令$d=p(a-n)-a$，那么$d\mid n(a-1)$。由于$p\ge 2$，所以$d\ge a-2n$。我们再令$a\le b$，可以得到$a=O(n)$。因为，如果$a>2n$，那么$a\le n+\frac{n(a-1)}{a-2n}$可以推出$a^2-4na+2n^2+n<0$，即$a\le 2n+\sqrt{2n^2+n}$。

于是我们预处理出$1\sim 2n+\sqrt{2n^2+n}$中每个数的质因数分解，然后枚举a，就可以在$O(\tau(n(a-1)))$的复杂度内得到答案，其中$\tau(x)$为x的因数个数。然而a较大的时候该算法较慢。当$a\le n+\frac{n(a-1)}{p(a-n)-a}$时，$p\le \frac{a^2-n}{(a-n)^2}$。当$a\ge n+3000$时$p\le 1178$，直接$O(np)$枚举即可。

时间复杂度：$O(n(p+\tau(O(n^2))))$

空间复杂度：$O(n)$

§97 MARTARTS

§97.1 题目描述

有一张完全二分图，每条边上有两个权值a,b。要求选出一个完备匹配。选出完备匹配之后，删去其中a_i-b_i最大的那条边，如果该边$a_i\le b_i$则不删；如果有多条那么删除b_i最小的那条。要求最大化剩下那些边中a_i-b_i的和。如果有多种方案那么最大化剩下那些边中a_i的和。

§97.2 数据范围

$1\le N\le 100$，$0\le a_{i,j},b_{i,j}\le 10^{12}$。

§97.3 解题报告

按照a_i-b_i从大到小枚举被删的是哪条边。对于a_i-b_i相同的所有边，a_i小的排前面。设删掉边e，即求**只考虑$a_i-b_i\le a_e-b_e$的边，强制e在匹配中的最大匹配**。

可以看成下面这个过程：开始所有边的边权为原来边权，每次考虑一条边，把它的边权变成$+\infty$(强制匹配)，求出匹配大小更新答案，然后把它的边权变为$-\infty$。求匹配大小的时候可以认为边权是一个二元组(a_i-b_i,a_i)，最大化这个二元组。

考虑用KM算法维护最大权匹配。左、右边点的顶标分别记为α_i，β_i。更改一条边(u,v)的权值，先把u和u的匹配边从匹配中删掉，然后修改u的顶标为$w(u,i)-\beta_i$的最大值，最后以u为起点在相等子图中跑匈牙利算法，调整顶标，直到相等子图中找到新的增广路。

注意边权中的$+\infty$要小于初始时$slack$的$+\infty$，不然会死循环。

再加上几个剪枝，如考虑的边i满足$a_i\le b_i$时用最大匹配更新答案，直接退出；无法更新答案的时候直接退出。这样就可以通过本题了。

时间复杂度：$O(N^4)$

空间复杂度：$O(N^2)$

§98 CNTHEX

§98.1 题目描述

求这样的六元组$(a_1,a_2,\ldots,a_6)$的个数模10^9+7：

1) $a_1\le a_2\le\cdots\le a_6$；
2) $\sum_{i=1}^{5}a_i>a_6$；
3) $1\le a_i\le N$；
4) $a_6\ge L$；
5) $a_1,\ldots,a_5\le X$；
6) 相等的a_i不超过K个。

§98.2 数据范围

$2\le N\le 10^9$，$2\le L\le N$，$N-L\le 100$，$1\le X<L$，$1\le K\le 5$。

§98.3 解题报告

枚举a_6，我们按照二进制位从高到低确定$a_1 \dots a_5$。

令$f_{i,j,k,l,m}$表示方案数，其中

- i表示考虑了第$32-i \sim 32$位；
- $j = \lfloor \frac{a_1+\dots a_5}{2^{32-i}} \rfloor - \lfloor \frac{a_6}{2^{32-i}} \rfloor$。注意到$j < -4$就不合法了，$j > 0$就总是合法的。所以令$j \in [-4, 1]$，$j = 1$表示所有$j \geq 1$的情况。
- k表示$\lfloor \frac{a_5}{2^{32-i}} \rfloor$和$\lfloor \frac{X}{2^{32-i}} \rfloor$的大小关系(等于或前者小于后者)。
- l表示$\lfloor \frac{a_1}{2^{32-i}} \rfloor$和0的大小关系。
- m是一个四位二进制数，表示$\lfloor \frac{a_i}{2^{32-i}} \rfloor$与$\lfloor \frac{a_{i+1}}{2^{32-i}} \rfloor$是否相等。

转移的时候枚举$a_1 \dots a_5$的第$32-i$位是多少，如果满足条件可以算出新的状态。

时间复杂度：$O((N-L)Y4^Y \log X)$，其中$Y = 5$。

空间复杂度：$O(Y2^Y \log N)$

§99 challenge LAND

§99.1 题目描述

有一个$N \times M$的整数矩阵A，其中有L个数$\in [1, 50]$，其他数都是0。现在要你给每个等于0的数赋上一个$[1, 50]$之间的值。最小化

$$\sum_{(i,j)\text{和}(k,l)\text{相邻}} 2^{|A_{i,j}-A_{k,l}|}$$

§99.2 数据范围

M, N从$[10, 100]$中随机选取，K从$[1, 10]$中随机选取，$L = \frac{MNK}{100}$。初始时没有两个相邻的非0格子；L个格子权值随机。

§99.3 解题报告

首先确定一个较好的初始解。我的方法是，对每个0格子(i, j)，求出每个非0格子(k, l)的值的加权平均，四舍五入取整。经试验，权值取$\frac{1}{(|i-k|+|j-l|)^2}$的时候效果较好。

然后每次随机一个本来是0的格子，设它相邻的格子有l个，权值分别为$a_1, \dots, a_l$，将它调成某两个权值的平均值，即$\frac{a_i+a_j}{2}$(如果不是整数，取上整和取下整都试一遍)。在这些解中选一个最优的作为这个格子的新权值。卡时即可通过此题。

时间复杂度：$O(NML + count \times 1)$(一次调整复杂度$O(1)$)

空间复杂度：$O(NM)$

§100 SIGFIB

§100.1 题目描述

给出整数M, N，求

$$\sum_{x+y+z=N} 6xyzf_xf_yf_z \bmod M$$

其中$f_i = i (i \leq 1), f_i = f_{i-1} + f_{i-2} (i > 1)$。

§100.2 数据范围

$0 \leq N \leq 10^{18}, 1 \leq M \leq 10^5$，多组数据，$\sum M \leq 10^6$。

§100.3 解题报告

首先牢记x,y,z是等价的。

$$
\begin{aligned}
\text{原式} &= \sum_{x+y+z=N} 6xy(N-x-y)f_xf_yf_z \\
&= 6N\sum_{x+y+z=N} xyf_xf_yf_z - 6\sum_{x+y+z=N} x^2yf_xf_yf_z - 6\sum_{x+y+z=N} xy^2f_xf_yf_z \\
&= 6N\sum_{x+y+z=N} xyf_xf_yf_z - 12\sum_{x+y+z=N} x^2yf_xf_yf_z
\end{aligned}
$$

记$A=\sum_{x+y+z=N} 6xyf_xf_yf_z$。那么

$$
\begin{aligned}
A &= \sum_{x+y+z=N} (2xy+2yz+2zx)f_xf_yf_z \\
&= \sum_{x+y+z=N} (N^2-x^2-y^2-z^2)f_xf_yf_z \\
&= N^2\sum_{x+y+z=N} f_xf_yf_z - 3\sum_{x+y+z=N} x^2f_xf_yf_z
\end{aligned}
$$

记$B=\sum_{x+y+z=N} 6x^2yf_xf_yf_z$。那么

$$
\begin{aligned}
B &= \sum_{x+y+z=N} (3x^2y+3y^2x)f_xf_yf_z \\
&= \sum_{x+y+z=N} ((x+y)^3-x^3-y^3)f_xf_yf_z \\
&= \sum_{x+y+z=N} (N^3-3N^2z+3Nz^2-x^3-y^3-z^3)f_xf_yf_z \\
&= N^3\sum_{x+y+z=N} f_xf_yf_z - 3N^2\sum_{x+y+z=N} xf_xf_yf_z + 3N\sum_{x+y+z=N} x^2f_xf_yf_z - 3\sum_{x+y+z=N} x^3f_xf_yf_z
\end{aligned}
$$

这样我们只需要求出$F_b(N)$，其中$F_b(N)=\sum_{x+y+z=N} x^bf_xf_yf_z$，$0\le b\le 3$。

Fibonacci数在模M意义下的循环节是$O(M)$的。令p为循环节长度。

考虑一个$f_xf_yf_z(x,y,z\in[0,p),x+y+z\equiv N \pmod p)$。它在$F_b(N)$中被加了$\sum_{i=0}^{t}(t-i+1)(ip+x)^b$次，其中$t=\frac{N-x-y-z}{p}$。这个$S(t,b)=\sum_{i=0}^{t}(t-i+1)(ip+x)^b$可以$O(1)$计算。只要得到$\sum_{i=0}^{t} i^b$的公式，对于$b=0\dots3$暴力拆式子就行。有点恶心。

我们令$g_t=\sum_{a=0}^{t} f_af_{t-a}$，那么$g_0=g_1=0,g_{t+1}=\sum_{a=0}^{t-1} f_a(f_{t-a}+f_{t-1-a})+f_t=g_{t-1}+g_t+f_{t-1}$。

令$h_t=\sum_{a+b=t,a,b\in[0,p)} f_af_b$，那么$t<p$时$h_t=g_t$；$t>p$时$h_t=g_t-\sum_{a=0}^{t-p} f_af_{t-a}-\sum_{a=p}^{t} f_af_{t-a}$。在模意义下它就等于$g_t-2g_{t-p}$。这两个函数在接下来的计算中有用。

枚举x，求$\sum_{y,z\in[0,p),y+z\equiv N-x \pmod p} S(\frac{N-x-y-z}{p},b)f_yf_z$。令$Q=N-x,q=Q \pmod p,r=\lfloor\frac{Q}{p}\rfloor$，答案等于$\sum_{y=1}^{q} S(r,b)f_yf_{q-y}+\sum_{y=q}^{p-1} S(r-1,b)f_yf_{q+p-y}=S(r,b)g_q+S(r-1,b)h_{q+p}$

整理一下算法流程: 预处理Fibonacci数列、循环节p、g数组; 计算$f_b(N)$的时候，枚举x，算出Q,q,r，将$F_b(N)$加上f_x乘$S(r,b)g_q+S(r-1,b)h_{q+p}$。然后把$F_b(N)$代入答案即可。

时间复杂度：$O(\sum M)$

空间复杂度：$O(\max M)$

§101 CIELQUAK

§101.1 题目描述

给定$R\times C$的网格图，每条边存在的概率是$1-p$，求$(1,1)$与(R,C)连通的概率。

§101.2 数据范围

$1\le R\le 8,1\le C\le 10^{18},0.1\le p\le 1$。

§101.3 解题报告

使用插头dp，复杂度是状态数S乘以矩阵的点数RC的。经过试验$R = 8$时$S = 3432$。预处理出所有状态和转移，dp的时候复杂度为$O(SRC')$。

可以证明当C比较大时，$\frac{ans_{R,C,p}}{ans_{R,C-1,p}}$趋近一个常数，取$C' = 50$，算到$ans_{R,C',p}$时估计出这个常数，然后直接求幂算答案即可。

时间复杂度：$O(TSRC')$

空间复杂度：$O(SRC')$

§102 challenge SEAPERM

§102.1 题目描述

给出一个长为N的数组$A_1, A_2, \ldots, A_n$，定义$f(i) = S - \sum_{r=i}^{j} A_r$，其中$j$是满足$\sum_{r=i}^{j} A_r \le S$的最大$j$。如果$A_i > S$那么$f(i) = S$。要求用一个置换打乱$A$的顺序来最小化$f(1) + f(2) + \cdots + f(k)$。

§102.2 数据范围

数据组数不超过10，$1 \le k \le N \le 2000$，$1 \le A_i \le 10^4$，$1 \le S \le 10^9$。

§102.3 解题报告

首先一种比较优的方案是将A升序排序。

然后考虑一种贪心：已经确定了$A_{k+1} \ldots A_n$。我们选出最小化$f(k)$的那个A_k，然后从剩下数中选出最小化$f(k-1)$的那个A_{k-1}，然后继续选下去直到选出A_1。一次这样的贪心是可以在$O(N \log N)$时间内完成的。具体地说，用平衡树维护还没有确定位置的所有数。维护指针$r = n$和$l =$要填的位置，每次要选一个数，就从平衡树中找出$\sum_{i=l+1}^{r} A_i + w \le S$的最大$w$，如果不存在就$r \leftarrow r - 1$，否则更新平衡树和指针。需要特判$A_i > S$的情况。

随机多次取最优解。

时间复杂度：$O(N \log N + count \times N \log N)$

空间复杂度：$O(N)$