趣味数（p1）

【题目简述】

判断 $[10, n], n \leq 100000$ 存在多少个趣味数。一个数从高位到低位是不下降序列，这个数就是趣味数。

【算法 1：枚举判断】

枚举 $[10, n]$ 的每个数字，然后判断这个数是否是趣味数。一个数最多有 5 位，时间成本很低。通过除以 10 的商和余数，就能把每一位的数字取出来。

与我无关（p2）

作者：function2

【题目简述】

给两个已知离散随机变量 X,Y，给出 n 个 X,Y 的线性组合 $Z_i=a_iX+b_iY$，任选 3 个，求至少两个不相关的概率。

【参考解法】

注意到 $\mathrm{Cov}(X,Y)=E(XY)-E(X)E(Y)$ 是内积，Z_i 看作是在以 X,Y 为基的放射坐标系下的点，可以考虑单位正交化后放到平面上解决。

设 $(X,Y)=(\alpha_1,\alpha_2)$，尝试正交化，令 $\beta_1=\alpha_1,\beta_2=\alpha_2-\frac{(\alpha_2,\beta_1)}{(\beta_1,\beta_1)}\beta_1$，有

$$(\alpha_1,\alpha_2)=(\beta_1,\beta_2)\begin{pmatrix}1 & \frac{(\alpha_2,\beta_1)}{(\beta_1,\beta_1)}\\ & 1\end{pmatrix}$$

再单位化，由 $(\beta_2,\beta_2)=(\alpha_2,\beta_2)=(\alpha_2,\alpha_2)-\frac{(\alpha_2,\alpha_1)^2}{(\alpha_1,\alpha_1)}$，令 $\gamma_1=\frac{1}{|\beta_1|}\beta_1,\gamma_2=\frac{1}{|\beta_2|}\beta_2$，有

$$(\alpha_1,\alpha_2)=(\gamma_1,\gamma_2)\begin{pmatrix}|\beta_1| & \\ & |\beta_2|\end{pmatrix}\begin{pmatrix}1 & \frac{(\alpha_2,\alpha_1)}{(\alpha_1,\alpha_1)}\\ & 1\end{pmatrix}=(\gamma_1,\gamma_2)\begin{pmatrix}|\beta_1| & |\beta_1|\frac{(\alpha_2,\alpha_1)}{(\alpha_1,\alpha_1)}\\ & |\beta_2|\end{pmatrix}=(\gamma_1,\gamma_2)C$$

那么设输入坐标为 x ，变换后为 y ，有

$$(\gamma_1,\gamma_2)y=(\alpha_1,\alpha_2)x=(\gamma_1,\gamma_2)Cx$$

由于 γ_1,γ_2 线性无关，因此 $y=Cx$ 。

不相关也就是正交，在变换后的平面上求一个直角三角形个数就可以了。

枚举直角顶点，把剩下的点按照极角排序，用两个指针求解直角数量即可。时间复杂度 $O(n^2\log n)$

因数分解（p3）

【题目简述】

- 在 k 维笛卡尔坐标系下，第 i 维坐标的取值为不超过 a_i 的正整数（$a_i \geq 2$）。总共有 $a_1a_2\cdots a_k$ 个点，每个点上有唯一的编号。
- 当且仅当两个点距离为 1 时（即恰好有一维坐标相差 1），这两点之间会连一条无向边，设共计有 m 条边。
- 输入：m 和每条边两个端点的编号。
- 求 k 与 $a_1, a_2, \ldots, a_k$。

【标准算法】

第一问

可以证明，度数最小的点是角，这个度数就是 k。

时间复杂度 $O(m)$。

第二问

- 选出一个角，令其坐标为 $(1, 1, \ldots, 1)$。
- 进行**广度优先搜索**，求出每个点到角的距离和前驱。
  - 对于距离为 1 的点，令其坐标某一维为 2，其余为 1；并保证这些点坐标不同。
  - 否则，如果它只有一个前驱，则令其坐标为前驱坐标在其最大一维上 $+1$。
  - 否则，令其坐标为各前驱坐标的最大值。
- 将最远点的坐标排序后，得到 $a_1, a_2, \ldots, a_k$。

显然，解是唯一的（想象空间中的 k 维立方体，解释方法一定是唯一的）。

瓶颈在于搜索，时间复杂度 $O(m)$。

【部分分算法】

其他的做法可以考虑到，由于 n（图的点数）与 m 是已知的，则可以暴力枚举所有可能的 $a_1, \ldots, a_k$ 然后依次代入检验（因为 $n = a_1a_2\cdots a_k$）。对于所有的 a 均为质数的子任务，这样分解完成后就可以直接输出答案了。

检验边数是否符合要求比较容易，问题在于检验两个图是否同构。一种可能的“乱搞”其实也是成立的：处理所有点对间的最短路然后排序，这样对于同构的图得到的结果一定是一样的。由于到此为止可能的图的种类已经不太多了，如此判断就可以了。

求所有点对间的最短路可以用 Floyd 算法或者从每个点开始搜索，得到不同的部分分。

幸运盒（p4）

【题目简述】

有 n 个初始状态为空的盒子，每次可以选择：
恰好 k 个概率为 $p\%$ 的盒子，进行一次重新随机；
恰好 t 个概率从 $p\%$ 开始递增的的盒子，进行一次重新随机；
不进行操作。
问进行 m 次操作后最大期望。

【算法：DP】

随着剩余次数和当前环境情况的不同，最优决策也不同，但是一共就三个操作，可以进行枚举而获取最优期望。

令 $dp_{i,j}$ 表示剩余 i 轮，目前有 j 个不是期望物品的盒子的状态下，获得最大期望糖果的数量，可以 $O\left(n^3\right)$ 对期望进行动态规划。

首先进行预处理，令 $P_{1,v}$ 和 $P_{2,v}$ 分别表示使用操作 和 操作，开出 v 个糖果的概率。可以 $O\left(k^2+t^2\right)$ 预处理出这两个概率。预处理思路如下：

令 $f_{i,j}$ 表示对前 i 个盒子进行随机后，获得 j 个糖果的概率是多少。$f_{0,0}=1$，$f_{i,j}=f_{i-1,j}\times(1-p)+f_{i-1,j-1}\times p$ 其中，p 表示第 i 个盒子的概率，对应步骤 即为 $p\%$，对应步骤 即为 $(\min(p+i-1,100))\%$ 。

初始状态：$dp_{0,j}=0, j=0,1,\ldots,n$ ，表示一开始什么都没有。

状态转移：

令 f_k 表示使用操作 进行随机可以获得的期望，枚举可能随机到期望状态的个数 v，就可以获知对应于状态 $dp_{i,j}$ 使用操作 的前一个状态；于是就可以用前一个状态的期望与操作 随机出 v 个糖果的增益的和，乘上随机出 v 个糖果的概率 $P_{1,v}$ ，进行状态转移。对于前一个状态，令：

$$s=\max(k,j)$$

则前一个状态即为 $dp_{i-1,s-v}$ ，开出 v 个糖果的增益即为 $j-s+v$ 。这是因为对于步骤 ，无论操作前有几个糖果，他都需要选择恰好 k 个糖果进行随机，很显然我们都要尽量选择非糖果的盒子进行操作才是最优的，如果操作前非糖果的盒子数 j 超过 k 的话，相当于先减少了 $j-k$ 个糖果，再考虑随机后的增益。增益值即为 $j-s+v$ 。

$$fk=\sum_{v=0}^{k}\left(f_{i-1,s-v}+j-s+v\right)\times P_{1,v}$$

那么同理，对于操作 的期望值 ft，有下述方程：

$$s = \max(t, j)$$

$$ft = \sum_{v=0}^{t} (f_{i-1,s-v} + j - s + v) \times P_{2,v}$$

上述两个步骤的期望，加上不进行操作的选择（因为可能目前糖果已经足够多，进行随机可能反而导致更劣的情况），就可以得出 $dp_{i,j}$ 的方程：

$$dp_{i,j} = \max(dp_{i-1,j}, fk, ft)$$

枚举每个状态，复杂度为 $O(nm)$ ，状态转移复杂度为 $O(k)$ ，总复杂度 $O(nmk)$。

最终答案即为 $dp_{m,n}$ 。

进阶法师（p5）

idea：吴凯路 (wuvin) 出题：吴凯路 (wuvin) 验题：

【题目简述】

给出 N 个点，两两连边，以每一条边为直径作圆，求圆的并的面积.

这本来该是上一场 codeplus 的题，但是在考试前几天，标算被验题人 hack 了，只好改题，就有了上一场那道一共一个人通过的题目。

本次修补了之前标算的 bug，重新拯救这道题。一来想继续推广计算几何，让大家理解计算几何的多变。二来想推广 codeplus，坚持打 codeplus 有惊喜哦！

【算法：旋转卡壳】

原问题是如果一个点被保护，当且仅当它到某两个点形成的角度大于等于 90 度。显然可行区域的边界可以视为，使用一个直角去卡这个点集，然后在直角旋转过程中，端点留下的路径。可以看出只保留凸包上的点，不会对本题答案产生任何影响。

使用一个直角进行旋转卡壳，容易证明在旋转过程中，两边卡着的点都是单向移动的。当确定了卡着的两个点的时候，显然可以推出这个直角端点，在旋转的过程中经过的是一段以这两个点的连线作为直径的圆的圆弧。

那么我们将边界分为多段圆弧，并在凸包中任取一点作为中心，然后依次计算每段圆弧与中心形成的图形的面积，之和即为本题答案。

对于每个圆弧与中心形成的图形，是一个弓形与一个三角形，弓形面积可以使用扇形减去对应连向圆心的三角形，而原来的三角形面积可以通过向量叉积计算。

总复杂度 $O(n \log n)$

坐标转换（p6）

邢健开

【题目简述】

对于一个 $2^n \times 2^n$ 的网格图，对每个格子用两种方式给出一个标号，一种是正常的从上到下，另一种是 Z 字形分型编码，现给你一种标号方式中某个格子的编号，要求转换为这个格子在另一种编码方式中的编号。

【子任务设计】

前 6 个点数据范围较小

另有 6 个点只要求进行某一种转换。

【算法 1：打表】

对于前 6 个点，可以将表打出来放在程序里。

【算法 2: Z 转换 R】

将给出的标号看作四进制数，从高位到低位看，每次将当前网格分成 4 份，则每一位决定这个格子在哪一块中，然后递归进入这个块，重复这个过程即可。最后可确定这个格子的横纵坐标，然后再算出在 R 编码中的标号就可以了。

【算法 3:R 转换 Z】

先确定给出格子的横纵坐标。和上述算法类似，每次将当前网格分成 4 块，通过横纵坐标判断这个格子属于哪一块，然后递归进入这个块，重复这个过程。确定格子属于当前网格的第几个块的时候，可以求出在这之前有多少个块已经在 Z 编码中遍历过了，累加到答案即可。

校门外的树（p7）

出题人：打杂的 8
验题人：姜淳誉

【题目简述】

给定长为 n 的数列 $\{a_i\}$，q 次计算区间的两两最大公约数之积。

$$1 \leqslant n, q, a_i \leqslant 1e5$$

【算法 1:】

每次询问枚举询问区间的每对数计算最大公约数，时间复杂度为 $O(qn^2)$。
能通过 Subtask1，期望得分 10。

【算法 2:】

或许有某种做法仅能通过 Subtask2，但是我没有写，期望得分 10。

【算法 3:】

对于所有 $\{a_i\}$ 都相等的情况，显然可以直接 $\log a_i$ 计算每个区间的答案，时间复杂度为 $O(q \log a_i)$。
能通过 Subtask3，期望得分 10。

【算法 4:】

考虑每一个质因子，显然他们的贡献是独立的。
枚举所有质因子，用快速幂计算对每个询问区间的贡献，设数列中所有出现过的质因子数量为 k，则时间复杂度为 $O(qk \log \max a_i)$。
能通过 Subtask3 和 Subtask4，期望得分为 35。

【算法 5:】

小于 $\sqrt{\max a_i}$ 的质因子一共有 65 个，可以直接暴力计算贡献，时间复杂度仍为 $O(qk \log \max a_i)$，这里 $k = 65$。
我们注意到，大于 $\sqrt{\max a_i}$ 的质因子，在每个数中最多只会出现一次，直接用莫队算法维护这部分质因子的答案，时间复杂度为 $O\left(q\sqrt{n}\right)$。
能通过所有 Subtask，期望得分 100。

如果你用别的方式（如平衡树）维护大于 $\sqrt{\max a_i}$ 的质因子，或者常数不够好，又或者莫队每次转移的时候用快速幂而非维护当前各项幂次，那么可能无法通过 Subtask6，只能得到 70 分。

祖玛（p8）

作者：function2

【算法 1：暴力】

设 $f(l,r,x,v)$ 表示当前待选区间为 $[l,r]$，上一个选的数为 v（-1 表示没有），且选了 x 个的最大收益，则有

$$f(l,r,x,v)=\max\begin{cases} x^2 & l>r\ X_{min}\le x\le X_{max} \\ f(l,i-1,0,-1)+f(i+1,r,x+1,a_i) & a_i=v \\ f(l,i-1,0,-1)+f(i+1,r,1,a_i)\ x^2 & a_i\ne v\ X_{min}\le x\le X_{max} \end{cases}$$

注意到 v 这一维是没有用的，如果存在的话始终等于 a_{l-1}，故可以去掉。

总时间复杂度为 $O(n^4)$，常数很小。