Problem 1

Task 1

按照题意模拟

Task 2

$2\sum_{i=0}^{p}\lfloor\frac{iq}{p}\rfloor=\sum_{i=0}^{p}\lfloor\frac{iq}{p}\rfloor+\lfloor\frac{(p-i)q}{p}\rfloor=(p+1)\cdot(q-1)+\sum_{i=0}^{p}[p\mid iq]=(p+1)\cdot(q-1)+\gcd(p,q)+1$

Problem 2

Task 1

把所有质数筛出来，背包 $O(n^4)$

Task 2

先枚举两个质数相加，得到每个值 x 通过两个质数的和能够有多少种方案得到它。然后再枚举两个"两个质数的和"求和判断是否和 N 相等，时间复杂度 $O(n^2)$ 。

Task 3

第一个步骤和**Task 2**一样，之后发现只要枚举一个"两个质数的和"进行对应的计数就好了，时间复杂度 $O(n)$ 。

Problem 3

Task 1

对于每次查询，暴力搜索到 n的路径。

Task 2

注意到每次转移的状态只和当前点u 以及从s 到当前点经过的所有边的 gcd的值 g。

所以很容易可以想到一个 dp.

定义$dis[u][g]$为从起点到 u点，当前经过的所有边的gcd 为 g的最短路。

转移时枚举u点的出边(u,v,w)，定义$g'=gcd(g,w)$。则转移方程如下:

$dis[v][g']=min(dis[u][g]+\frac{w}{g'})$

由于每个点的状态数只与边的权值的因数有关，故总状态数为$m\times\sqrt{m}$ 。dp 的转移借助dijkstra 进行，一次查询的复杂度为$m\times\sqrt{m}\times\log n$ ， Q次查询的复杂度为 $Q\times m\times\sqrt{m}\times\log n$ （最后一个log 为dijkstra的复杂度），预计得分 60

Task 3

很容易注意到，终点是固定的，所以可以从这里下手。

但是有一点不是很方便，就是边权的定义。但边权只与起点到这条边的 gcd有关，因此可以枚举这个 gcd的值。

定义$dis[u][g]$ 为从u 点到终点，从起点到u 点的所有路径的 gcd为 g。

转移时枚举v 点的出边 (u,v,w),枚举g'的倍数g.注意到该转移方程只可在$g'|w$ 的时候进行转移

$dis[u][g] = min(dis[v][g'] + \frac{w}{g'})$

虽然需要枚举g，但该算法可以 $O(1)$地回答每个询问

该算法复杂度为$O(m \times V \times \log V + Q)$ （最后一个$\log$ 为dijkstra 的复杂度），足以通过此题

Problem 4

本题涉及了关于线性筛、质因数分解、置换、分析数据范围特性等多种技巧，若之前接触过则实际难度并不会超过**T3**。

Hint 1

可以把每个 p_i 看成从 i 连向 p_i 的一条有向边，这样整个图会由若干个互不相交的简单环构成（所有点的入度出度均为 1 ）。

可以通过分析得出， P^k 的意义就相当于每个人一开始都在初始的 i 号结点， k 每次 $+1$ 就变成所有人在图上走一步。

于是我们得出了 $f(i,j) = 0$ 的充要条件： i, j 在同一个环上。

Hint 2

考虑对一个排列 P 如何计算 $v(P)$ ，设 i 号结点所在的环长为 r_i ，则可以得出 $v(P) = LCM(r_1, r_2, \cdots, r_n)$ ，这是因为所有点都同时回到原点需要保证每个人的步数都为 r_i 的倍数。

接着考虑 $v(A_{ij})$ 的值，草稿纸上画一画就可以发现，把两个不同环上的点的出边进行交换，就会把两个环合并，因此 $v(A_{ij})$ 的值就为 $r_i + r_j$ 与其他环长取 LCM 的值。

到这一步实际上我们发现如果枚举 i, j 并暴力计算 LCM ，那么可以得到一个看上去是 $O(n^3 \log n)$ 的做法。

Hint 3

对上一步的暴力做法进行些许优化，对每个相同的 r_i 其实是不用重复计算的，所以设环长的种类数为 m ，能做到 $O(m^3 \log n)$ 的复杂度。

这里需要进行对数据范围分析，注意到这里是 m 个互不相同的数相加不超过 n ，所以一定有 $\sum_{i=1}^{m} \le n$ ，计算得出 m 是 $O(\sqrt{n})$ 级别的，所以时间复杂度**貌似**可以优化到 $O(n\sqrt{n}\log n)$ 。

Hint 4

但是我们会发现 LCM 的值可能会很大并不能直接计算，所以需要对每个 r_i 进行质因数分解，最后在每个质因子上计算指数的最大值得出结果，这样单次计算 LCM 的时间复杂度为 $O(m\sqrt{n} + cnt_{prime} \cdot \log\log n)$ 。

这里可以利用线性筛的性质来加速质因数分解的过程，注意到每次线性筛筛到一个数时，一定是被他的最小质因子筛到，于是可以预处理记录下每个数字对应的最小质因子，这样单次质因数分解的时间复杂度可以优化到 $O(\log n)$ 。

除此之外，我们还可以在每次分解质因数时直接质因子判断对应指数的值，并判断是否乘上去，如果采用 map 或 set 来记录，这样单次计算 LCM 的复杂度变成了 $O(m\log^2 m)$ ，总时间复杂度为 $O(n\sqrt{n}\log^2 n)$ 。

如果在判断每个质因子对应的最大指数时做到 $O(1)$ 的维护，那么就是 $O(n\sqrt{n}\log n)$ 的时间复杂度，期望得分80，如果能做到常数优秀且熟练掌握卡常技巧是完全有机会通过此题的。

Hint 5

继续优化求 LCM 的方法，我们可以预先求出所有数字的 LCM ，然后记录每个质因子对应的三个最大指数（因为后面要删掉两个数），这样每次删数的时候就可以现场质因数分解 $O(\log^2 n)$ 维护 LCM ，添加一个数字的时候也能够当场维护。

另外，也可以直接用 $multiset$ 记录每个质因子对应的指数集合，同样也是 $O(\log^2 n)$ 的维护复杂度。

总时间复杂度 $O(n\log^2 n)$ ，已经足够通过此题。