FAQ

Возможно (скорее всего), в некоторых (многих) задачах есть решение круче, оптимизированней и в 2 строки, но меня это не особо волнует.
По запросу в лс объяснение любой задачи может быть уточнено, главное пишите, что конкретно осталось неясно.
Enjoy

Task A

Отсортируем отдельно девочек и мальчиков по координате. $dp[i][j]$ – ответ, если были рассмотрены первые i мальчиков и j девочек.
Пересчёт: для состояния i, j переберём b, g – кол-во мальчиков и девочек в очередной машине ($b > 0 \wedge b \geq i \wedge g \geq j \wedge b+g \leq 4$). $dp[i][j] = \min(dp[i][j], dp[i-b][j-g]+maxcord)$.
Асимптотика $O(n^2)$.

Task B

Учебная задача, в интернете описаны все возможные способы решения (дп + бинпоиск, дп + ДО).
Асимптотика $O(n \log n)$.

Task C

Насчитаем массив p префиксных сумм на входных данных. $dp[i][j]$ – ответ, на задачу, если всего расставлено j банд и последняя из них находится на месте i-ого дома. Сначала посчитаем все значения $dp[i][1]$, то есть ответ, если есть только 1 банда на позиции i.

$$dp[i][1] = i \cdot homes[i] - pref[i] + pref[n] - pref[i] - (n-i) \cdot homes[i]$$

Теперь перебираем кол-во cnt банд с 2, потом позицию последней банды i, также позицию предыдущей банды j ($j \geq cnt - 2 \wedge j < i$). Найдём середину между ними – $m = \frac{homes[i]+homes[j]}{2}$. Бинпоиском по исходному массиву найдём ind позицию последнего дома, который будет относиться к банде с координатой $homes[j]$, и пересчитаем $dp[i][j]$. (Я не буду расписывать итоговые формулы, ибо антиплагиат). Схематично: все дома $[0; j]$ не меняют принадлежности бандам, $[j+1, ind]$ остаются у дома j, $[ind+1; n]$ уходят дому i.
Асимптотика $O(n^2 \log n)$.

Task D

Главная идея: давайте делать $dp[i][j]$ – максимальная высота здания, которую можно узнать имея i самолётов и j попыток. Тогда достаточно найти первое j, такое, что $dp[k][j] \geq n$.
Пересчёт:

$$dp[i][j] = dp[i][j-1] + dp[i-1][j-1] + 1$$

При этом если $\min(i, j) = 0 \Rightarrow dp[i][j] := 0$ и если $i = 1$, то $dp[1][j] := j$.
Асимптотика $O(k \cdot ans)$, при этом ans достаточно маленькое.

Task E

$dp[sum][last]$ – количество множеств с суммой sum и максимальным числом $last$. Также для каждого слоя динамики будем поддерживать $p[sum][i]$ префискcные суммы.
Пересчёт: перебираем sum, $last$; $prev = last/2$;

$$dp[sum][last] += p[sum - last][prev + 1]$$

И после
Пересчёт очередного слоя обновим $p[sum]$.
Асимптотика $O(n^2)$.

Task F

Главная идея: все числа различны, а следовательно длина ответа $\leq \log(1e18)$. $dp[len][i]$ – самое дальнее место начала цепочки длины len с концом в позиции i. Для удобства работы с циклическим массивом допишем к исходному ещё один такой же. Теперь, чтобы не зайти по циклу слишком далеко достаточно поддерживать инвариант $i - dp[len][i] < n$, иначе $dp[len][i]$ недостижима.
Пересчёт нужно делать только через позиции, которые делят наше $A[i]$.
Асимптотика $O(\log_2 1e18 \cdot n^2)$, что формально много, но при аккуратном написании работает $\leq 120ms$.

Task G

Будем хранить $dp[i][0, 1]$ – длина максимальной подпоследовательности, заканчивающаяся в i-ой позиции, причём последний знак $<, >$.
База: $dp[i][0] = 1, dp[i][1] = 1$.
Пересчёт: для i пройдём по всем $j < i$, если $a[j] < a[i]$, то $dp[i][0] = \max(dp[i][0], dp[j][1] + 1)$, иначе $dp[i][1] = \max(dp[i][1], dp[j][0] + 1)$.
Асимптотика $O(n^2)$.

Task H

Учебная, НОП.
Асимптотика $O(n^2)$.

Task I

Тык.
Асимптотика $O(nm)$.

Task J

Пусть now – количество различных подпоследовательностей, а $dp[x]$ – кол-во подпоследовательностей, оканчивающихся на x. Тогда для очередного $A[i]$:

$$b = now - dp[A[i]], dp[A[i]] = now, now += b$$

Асимптотика $O^*(n)$ (хеш-таблица).

Task K

Учебная, рюкзак.
Асимптотика $O(nM)$

Task L

Предполагается рюкзак, у меня другое решение и лучше я не буду его сюда писать...

Task M

Главная идея: для пары (аэропорт, прямая) ответ это $w \cdot F_{dst}$, где w – кол-во рейсов до аэропорта (из условия), F_i – i-ое число Фибоначчи, dst – расстояние от аэропорта до прямой.
Просто перебираем пару (аэропорт, прямая). Если аэропорт расположен ниже прямой, SKIP. Предположим что через аэропорт проведена прямая $y = x + b$, найдём $b = airport_y - airport_x$, тогда $dst = b - c$. Теперь найдём dst-ое число Фибоначчи через возведение матриц в степень и добавим к ответу $k_i \cdot F_{dst}$.
Асимптотика $O(nq \cdot 2^3 \cdot \log 10^{18})$.

Task N

Главный приколыч: давайте забьём на замену символов и будем считать какую-то динамику только от кол-ва добавлений и удалений, а потом, получив самые близкие к β строки для всех вариантов, за оставшиеся действия сделаем замены.
$dp[i][d][a]$ – мы из какого-то префикса строки α сделали строку длины i, применив d удалений и a добавлений. А значение динамики – кол-во совпавших позиций с префиксом β длины i.
База: $dp[0][0][0] = 0$.
Пересчёт: если $d > 0$, $dp[i][d][a] = \max(dp[i][d][a], dp[i][d-1][a])$, если $a > 0$, $dp[i][d][a] = \max(dp[i][d][a], dp[i-1][d][a-1]+1)$, если $i - a + d > 0$ (это просто длина префикса α, из которого получена строка длины i), то $dp[i][d][a] = \max(dp[i][d][a], dp[i-1][d][a] + \alpha[i-a+d-1] == \beta[i-1])$.
После насчёта динамики пройдём по всем значениям $dp[|\beta|][a][d]$, и к каждому добавим $k-a-d$, так как ровно столько замен у нас осталось. Выберем максимальное состояние, это и будет ответ.
Асимптотика $O(nk^2)$.

Task O

Классическая учебная задача поиска гамильтонова пути минимальной длины, Google.
Асимптотика $O(n^2 \cdot 2^n)$.

Task P

Предполагалось: дп по изломанному профилю. Чит: пусть зафиксирована маска $mask$ i-ого столбца таблицы, тогда по ней и первому числу предыдущего столбца однозначно восстанавливается маска предыдущего столбца, а следовательно обновить $dp[i][mask]$ можно максимум из двух состояний.
Пересчёт: $dp[i][mask] += dp[i-1][prevmask]$
Асимптотика $O(2^n \cdot m)$.

Task Q

$dp[mask]$ – максимальное кол-во пар, вершин из маски, которые соединены ребром.
Пересчёт: перебираем две вершины u, v из маски, которые к тому же соединены ребром, и пытаемся обновить $dp[mask]$ через $dp[mask \oplus 2^u \oplus 2^v]$.
Асимптотика $O(2^n \cdot n^2)$.

Task R

Выберем из двух сторон меньшую. Пусть НУО это m. Тогда $dp[i][mask]$ – ответ для первых i столбиков, последний равен $mask$.
Пересчёт: перебираем дополнительно предыдущую маску, проверяем, что нет квадратов 2 на 2 одного цвета за $O(m)$ и добавляем текущему состоянию значение $dp[i-1][prevmask]$. Ответ – сумма всех чисел в последнем столбце.
Асимптотика $O(2^{2m}nm)$, так как $nm \leq 30$, то $m < 6$.

Task S

Вся динамика аналогична предыдущей задаче с той лишь разницей, что из за ограничений на n нам нужно воспользоваться возведением матрицы в степень, так как все переходы статичны. То есть $matr[prevmask][mask] = 1$, если переход как в прошлой динамике возможен. Далее возводим эту матрицу в степень $n-1$ с помощью BinPow. Домножим на строку из единиц соответствующего размера. Ответ – сумма всех чисел в итоговой матрице. Для упрощения жизни лучше написать перевод n в двоичную систему счисления.
Асимптотика $O((2^m)^3 \cdot \log n)$.

Task T

Тупая динамика – $dp[i][j]$ – кол-во способов добраться до клетки (i,j). $dp[i][j] = dp[i-1][j-1] + dp[i-1][j] + dp[i-1][j+1]$, для граничных может не быть члена $dp[i-1][j-1]$ или $dp[i-1][j+1]$. То есть все переходы статичны, а следовательно Воспользуемся возведением матрицы в степень: $matr[i][j] = 1$, если $|i-j| \leq 1$. Тогда для одного отрезка если мы знаем кол-во способов добраться до каждой клетки первого столбца, возведя матрицу переходу размера c_i^2 в степень $b_i - a_i$, и, домножив на этот столбец, получим последний.
Асимптотика $O(nc^3 \log 1e18)$.

Task U

Воспользуемся техникой MITM (meet-in-the-middle). Разобьём наши вершины на 2 равные половины просто по номерам. Теперь давайте в каждой половине по отдельности проверим для каждой маски (подмножества), является ли она кликой. Для этого будем поддерживать ещё один массив $andmask[mask] = g[i_1]\&g[i_2]\&...\&g[i_k]$, $i_1, ..., i_k \in mask$. Также нужно поддерживать $g[i]$ как маску вершин, с которыми соединена i-ая. Теперь для $mask$, $isclique[mask] = true$, если

$$andmask[mask]\ \&\ mask == mask$$

При этом $v = \log(mask \,\&\, -mask)$, наименьший номер вершины, входящей в $mask$.

$$andmask[mask] = andmask[mask \oplus 2^v] \,\&\, (g[v] \oplus 2^v)$$

Таким образом мы умеем подсчитывать $isclique$ для обеих половин за $O(2^{\frac{n}{2}})$. Кроме того, для левой половины давайте насчитаем f, где $f[mask] = \sum\limits_{mask' \subseteq mask} isclique[mask']$. С его помощью ответ восстанавливается легко: пройдём по всем маскам правой половины, являющихся кликами, и добавим к ответу $1 + f[and_{left}]$, где and_{left} – это по сути те вершины, из которых можно дополнить нашу $mask$ до новых, больших клик, опять же за $O(2^{\frac{n}{2}})$. Теперь осталось научиться считать сумму по всем подмаскам, а это известная техника SOS DP
Асимптотика $O\left(\frac{n}{2} \, 2^{\frac{n}{2}}\right)$.