

Побег (160 мин., 200 баллов)

Совершён побег из тюрьмы. В ходе расследования обнаружилось, что беглец знал некоторые части планировки здания и нанёс их в виде татуировки на своё тело. Фотографию татуировки отсканировали и сделали цифровую обработку.

Татуировка, а так же каждая из частей планировки здания представляет собой рисунок. Выяснилось, что каждый рисунок состоит из вертикальных, горизонтальных и диагональных ($+45^\circ$, -45° от вертикали) отрезков. Чтобы скрыть информацию о частях планировки, на татуировку также были нанесены различные дополнительные отрезки.

Так как части планировки имеют разные размеры, то при нанесении на татуировку они независимо друг от друга могли быть изменены в масштабе (уменьшены или увеличены), могли быть повернуты на 45° влево или вправо от вертикали, также они могли быть наложены друг на друга.

Любой рисунок задаётся так, что соседние отрезки (имеющие общую вершину) не лежат на одной прямой, а также любые два отрезка не имеют общего сегмента. В рисунках нет отрезков нулевой длины. На любом рисунке отрезки могут также образовываться в результате пересечения исходных отрезков.

Если татуировка включает часть планировки здания, то каждому отрезку этой части соответствует отрезок на татуировке.

Необходимо определить, какие части планировки здания включены в татуировку.

Входной файл

Входной файл *input.txt* содержит описания рисунков. Первый рисунок – татуировка, все остальные – части планировки здания (не более 20). Каждый рисунок описывается следующим образом. В начале описания следует строка, в которой записано положительное целое десятичное число n (не более 30), равное количеству отрезков, описывающих рисунок. Затем записаны n строк, каждая из которых определяет один отрезок. Описание отрезка – это четыре целых десятичных числа x_1 , y_1 , x_2 и y_2 (разделитель – пробел), соответствующие координатам вершин отрезка $(x_1, y_1) - (x_2, y_2)$. Все координаты принадлежат диапазону от 0 до 100 включительно.

Последняя строка не заканчивается символом перевода строки.

Выходной файл

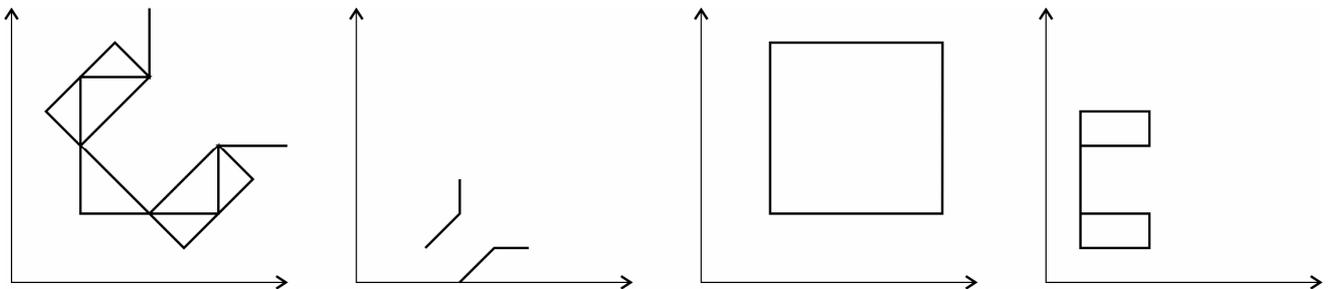
Выходной файл *output.txt* содержит одну строку, в которой записаны числа 0 или 1 (разделитель – пробел). Каждое число определяет присутствие (1) или отсутствие (0) соответствующей части планировки на татуировке.

Выходной файл не должен заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
<pre> 13 2 2 2 6 2 6 4 6 2 2 6 2 6 2 6 4 1 5 5 1 1 5 3 7 5 1 7 3 2 4 4 6 4 2 6 4 4 6 4 8 6 4 8 4 7 3 6 4 4 6 3 7 4 3 0 4 1 4 1 5 1 3 3 3 2 3 2 2 1 4 2 2 7 2 2 2 2 7 7 2 7 7 2 7 7 7 7 1 1 3 1 1 2 3 2 1 4 3 4 1 5 3 5 1 1 1 5 3 1 3 2 3 4 3 5 </pre>	<pre> 1 0 1 </pre>

Рисунок слева – татуировка, следующие три рисунка – части планировки здания. Первая и третья части планировки здания присутствуют на татуировке, вторая часть планировки здания – нет.





Камни (30 мин., 50 баллов)

Необходимо разложить несколько камней на две чаши весов так, чтобы отличие между массами, которые находятся на чашах, было минимальным.

Входной файл

Входной файл *input.txt* состоит из одной строки, которая содержит разделённые пробелом целые десятичные числа (не более 60 в диапазоне от 1 до 1000 включительно), каждое из которых равно массе одного камня.

Эта строка не заканчивается символом перевода строки.

Выходной файл

Выходной файл *output.txt* содержит одно целое десятичное число — абсолютная величина разности масс, положенных на чаши весов.

Выходной файл не должен заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
6 7 1 3 2	1

Агентство (90 мин., 100 баллов)

Вы работаете в брачном агентстве. У вас есть информация о женихах и невестах. Для эффективной работы агентства вы должны подобрать максимальное количество пар мужчин и женщин, которые подходят друг другу.

Количество мужчин n (не более 50) равно количеству женщин.

Исходная информация задаётся в виде квадратной матрицы, где каждая ячейка с координатами (i, j) принимает значение 0 или 1. Значение 1 обозначает, что женщина с порядковым номером j и мужчина с порядковым номером i подходят друг другу, 0, если не подходят.

Входной файл

Входной файл *input.txt* состоит из строк. В первой строке задано положительное целое десятичное число n . В следующих n строках расположены строки матрицы, в каждой строке записаны n чисел (0 или 1), разделитель – пробел.

Последняя строка не заканчивается символом перевода строки.

Выходной файл

Выходной файл *output.txt* содержит одно целое десятичное число, равное максимальному количеству подобранных пар.

Выходной файл не должен заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
4 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 0	4



Распознавание подписей (160 мин., 200 баллов)

Банку нужна программа, которая позволяет распознавать подписи клиентов на чеках. После сканирования и предварительной обработки каждая подпись представляет собой ломанную, для которой известна начальная точка. Распознавание подписей производится на основе их сравнения с образцами.

О том, насколько ломанные (образец и подпись) близки друг к другу, можно судить исходя из сравнения длин соответствующих сегментов образца и подписи, а также сравнения величин соответствующих углов между смежными сегментами. Если количество сегментов в образце и подписи не совпадает, то считают, что длины недостающих сегментов и величины соответствующих углов равны нулю.

Входной файл

Входной файл *input.txt* в первой строке содержит целое десятичное число C ($0 < C \leq 50$), которое определяет количество образцов подписей. Далее следует $(C * 2)$ строк, которые описывают образцы подписей.

Описание одного образца состоит из двух строк. Первая строка содержит имя клиента (не более 255 строчных букв английского алфавита), вторая строка описывает подпись (ломанную) и содержит пары (не более 100) целых чисел в диапазоне от 0 до 32000 включительно (разделитель между парами – пробел). Числа в паре разделены пробелом, первое число – координата X , второе – координата Y вершины ломанной. Первая пара определяет начальную точку.

Далее во входном файле следует строка, которая содержит целое десятичное число P ($0 < P \leq 50$) – количество распознаваемых подписей. Затем следуют P строк, в которых описываются подписи клиентов (аналогично описанию подписи в образце).

Последняя строка не заканчивается символом перевода строки.

Выходной файл

Выходной файл *output.txt* имеет следующий формат. В первой строке записано целое десятичное число P . Затем следует P строк, каждая из которых содержит имя клиента, которому принадлежит распознанная подпись. Последовательность имён в выходном файле соответствует последовательности подписей клиентов из входного файла, которые надо распознать.

Последняя строка не должна заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
<pre> 3 west 22 22 29 80 35 54 44 76 53 19 58 64 77 49 61 47 62 70 103 47 83 47 96 61 82 66 123 13 115 64 133 54 97 37 136 36 little 34 119 25 182 55 161 50 180 88 120 72 180 91 169 62 146 118 122 108 176 128 164 98 151 154 122 147 119 140 169 172 150 154 147 156 165 179 165 smith 79 257 39 285 83 314 44 343 100 308 95 337 116 307 115 332 135 305 137 332 154 304 156 332 182 254 173 331 198 318 146 292 221 253 216 328 242 299 242 325 6 369 21 372 75 386 44 397 69 408 15 422 52 439 46 416 35 426 62 480 36 450 38 475 57 455 61 510 5 499 59 523 47 481 21 528 28 382 101 372 162 395 136 393 161 426 101 419 162 441 151 399 121 477 96 462 159 493 146 446 113 526 105 511 96 509 152 534 132 519 129 518 151 533 151 354 224 349 264 391 255 389 291 401 241 413 268 414 230 431 253 432 220 450 241 450 216 461 228 442 184 471 228 479 214 436 207 465 173 494 213 495 182 508 198 612 385 643 348 609 325 629 288 591 319 591 287 573 316 569 281 552 319 547 285 531 317 532 286 502 358 503 288 480 298 527 325 465 364 461 291 447 318 438 290 678 307 680 345 714 334 709 364 725 323 736 344 737 312 747 326 750 301 763 317 774 284 780 309 806 248 800 307 817 301 780 272 857 264 835 312 866 301 855 327 593 129 589 150 597 143 597 150 607 125 605 149 613 143 597 135 624 125 620 147 628 140 615 129 639 126 634 122 634 147 645 134 637 134 640 146 646 144 </pre>	<pre> 6 west little smith smith little </pre>

west


west


little


little


little


smith


smith




smith


Образцы подписей

Подписи

Числа Фибоначчи (40 мин., 60 баллов)

Любое целое положительное число N может быть представлено единственным образом в виде:

$$N = F_{k_1} + F_{k_2} + \dots + F_{k_m}$$

где F_{k_j} – это положительное число из последовательности чисел Фибоначчи, а также выполняется условие:

$$k_1 \gg k_2 \gg \dots \gg k_m$$

которое означает, что k_j превышает k_{j+1} больше чем на 1.

Числа Фибоначчи – это последовательность чисел $F_0, F_1, F_2, F_3, \dots$, таких, что $F_0 = 0, F_1 = 1, \dots, F_i = F_{i-1} + F_{i-2}$ (для $i > 1$).

Входной файл

Входной файл *input.txt* содержит целое десятичное число N ($1 \leq N < 10^{39}$).

Входной файл не заканчивается символом перевода строки.

Выходной файл

Выходной файл *output.txt* состоит из одной строки, которая содержит упорядоченные по убыванию положительные десятичные целые числа F_{k_j} , разделитель – пробел.

Выходной файл не должен заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
1000000	832040 121393 46368 144 55

Ковёр (60 мин., 150 баллов)

Технология изготовления ковра основана на правилах:

1. Берут n ниток, каждая из которых имеет длину L , из которых будет изготавливаться ковёр.
2. Ковёр изготавливают рядами.
3. При изготовлении любого ряда используют два приёма: первый – это протягивание нитки через ряд, при этом расходуется c_1 её длины; второй – это связывание двух соседних ниток, при этом расходуется $(c_1 + c_2)$ длины каждой из них и они меняются местами. В одном ряду нитка может связываться только один раз.
4. Весь ряд можно изготовить только в том случае, когда для его изготовления хватает длины всех ниток.
5. Ковёр – это такая последовательность рядов, в которую нельзя добавить любой другой ряд.
6. В готовом ковре оставшиеся концы ниток обрезаются.
7. Нитку можно протягивать не более, чем через k рядов, но в каждом ряду должны быть связаны хотя бы какие-либо две соседние нитки.
8. Ковёр не должен распадаться на несвязанные части.

Надо определить количество возможных рисунков ковра (зеркально симметричные рисунки относительно вертикальной и горизонтальной осей симметрии считаются одинаковыми).

Входной файл

Входной файл *input.txt* состоит из одной строки, которая содержит пять целых десятичных чисел L , n , k , c_1 и c_2 , разделитель – пробел.

Эта строка не заканчивается символом перевода строки.

Гарантируются следующие диапазоны значений: $1 \leq L \leq 17$, $2 \leq n \leq 5$, $1 \leq k \leq 3$, $c_1 \geq 2$, $c_2 \geq 1$.

Выходной файл

Выходной файл *output.txt* содержит одно целое десятичное число, которое равно количеству возможных рисунков ковра.

Выходной файл не должен заканчиваться символом перевода строки.

Пример

<i>input.txt</i>	<i>output.txt</i>
9 4 2 2 1	9

Для этих входных данных существует девять рисунков ковра:

