

А. Ханты-Мансийск – Париж

Имя входного файла:	sms.in
Имя выходного файла:	sms.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт
Максимальная оценка:	100 баллов

В 2050 году руководство Глобальной Телефонной Сети (ГТС) приняло решение о новой системе тарификации коротких текстовых сообщений. Теперь цена отправки одного сообщения зависит от количества совпадающих цифр в начале номеров телефонов отправителя и получателя. Если первые c цифр телефонов совпадают, а $(c + 1)$ -я цифра различается, то стоимость сообщения составляет $(10 - c)$ кредитов ($0 \leq c \leq 9$). Все номера телефонов — десятизначные. При этом ГТС разрешает каждому абоненту отправлять сообщение только в пределах часового пояса своего проживания или часовых поясов, отличающихся от него на 1 час.

Школьник Поликарп из Ханты-Мансийска (время +2 часа от московского) успешно решил все задания первого тура олимпиады школьников по информатике. Теперь он желает сообщить об этом в Париж (время -2 часа от московского) своему учителю — профессору де Кодеру. Так как Ханты-Мансийск и Париж находятся не в соседних часовых поясах, Поликарп не может послать сообщение напрямую. Поэтому он пользуется тем, что у него есть друзья, которые проживают в Ханты-Мансийске, Париже, а также в промежуточных часовых поясах — в Дубае (время +1 час от московского), Москве и Калининграде (время -1 час от московского). Друзья Поликарпа по цепочке доставят профессору де Кодеру столь важную информацию. Поликарп хочет организовать передачу информации таким образом, чтобы минимизировать суммарные расходы по отправке всех сообщений.

Напишите программу, определяющую цепочку доставки, для которой суммарная стоимость отправленных сообщений минимальна.

Формат входных данных

Первые две строки входного файла содержат телефонные номера Поликарпа и профессора де Кодера. Далее следуют 5 блоков данных, описывающих друзей Поликарпа, живущих в Ханты-Мансийске, Дубае, Москве, Калининграде и Париже, соответственно. Каждый блок начинается со строки, содержащей одно число n_i ($1 \leq n_i \leq 100\,000$) — количество друзей Поликарпа в соответствующем городе, после которой следуют n_i строк — номера телефонов друзей. Все номера телефонов состоят ровно из 10 цифр. Гарантируется, что сумма всех n_i не превосходит 100 000. Все номера телефонов во входных данных различны.

Формат выходных данных

В первой строке выходного файла выведите минимальную возможную стоимость передачи информации w и количество задействованных в цепочке телефонных номеров k . Далее выведите k номеров телефонов, описывающих саму цепочку, в порядке следования от Поликарпа к профессору де Кодеру. Первый номер в цепочке должен совпадать с номером телефона Поликарпа, а последний — с номером телефона профессора де Кодера. Если решений несколько, выведите любое.

Примеры входных и выходных данных

sms.in	sms.out
1000000000 5000000000 1 9999999999 1 2000000000 1 3000000000 1 4000000000 1 8888888888	40 5 1000000000 2000000000 3000000000 4000000000 5000000000
2358847598 0023483473 1 0454385729 2 2358847500 2358840000 2 2358840001 2358847501 1 2358840002 1 0023483471	16 5 2358847598 2358840000 2358840001 2358840002 0023483473

Система оценивания

- Решения, корректно работающие при сумме n_i , не превосходящей 500, будут оцениваться из 40 баллов.
- Решения, корректно работающие при сумме n_i , не превосходящей 5000, будут оцениваться из 60 баллов.

В. А олени — лучше!

Имя входного файла:

deer.in

Имя выходного файла:

deer.out

Ограничение по времени:

2 секунды

Ограничение по памяти:

256 мегабайт

Максимальная оценка:

100 баллов

Вы уже знаете, сколько нефти добывается в Ханты-Мансийском автономном округе. Другой хозяйственной отраслью Югры является оленеводство. Нередко можно увидеть, как на нефтяной площадке, окружённой изгородью, работают нефтяники, а вокруг изгороди пасутся олени.

Оленевод Ванхо привязал своего оленя Ахтамака к изгороди нефтяной площадки, имеющей форму выпуклого многоугольника. Олень был привязан на длинной верёвке, чтобы он не убежал и при этом мог пастись. Вокруг нефтяной вышки растёт такой вкусный ягель, что олень тут же принялся его щипать.

Напишите программу, вычисляющую площадь участка вне изгороди, ягель на которой будет доступен оленю. Форма изгороди, точка привязывания и длина верёвки задаются во входном файле.

Формат входных данных

В первой строке входного файла записано целое число n — количество углов изгороди ($3 \leq n \leq 100$). В последующих n строках записаны координаты углов изгороди в порядке обхода по часовой стрелке. В последней строке записаны три числа — координаты точки привязывания оленя к изгороди и длина верёвки. Все координаты целые и не превосходят по модулю 10^4 . Длина верёвки — целое положительное число, не превосходящее 10^4 . Числа в каждой строке разделены пробелами. Гарантируется, что изгородь представляет собой строго выпуклый многоугольник и точка привязывания оленя лежит на его границе.

Формат выходных данных

В выходной файл выведите значение площади с точностью не менее 10^{-3} .

Примеры входных и выходных данных

deer.in	deer.out
4 0 0 0 2 4 2 4 0 1 2 2	7.06858
3 0 0 0 1 1 0 0 0 2	11.49557428756

Система оценивания

Решения, корректно работающие на тестах из примеров, а также в случае, если длина верёвки не превосходит половины периметра изгороди и изгородь представляет собой прямоугольник со сторонами, параллельными осям координат, будут оцениваться из 30 баллов.

Решения, корректно работающие на тестах из примеров, а также в случае, если длина верёвки не превосходит половины периметра изгороди, будут оцениваться из 60 баллов.

С. Земледелие 2.0

Имя входного файла:	farmer.in
Имя выходного файла:	farmer.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт
Максимальная оценка:	100 баллов

Фермер Архип решил заняться земледелием и выращивать брюссельскую редиску. Для этого он купил прямоугольное поле, состоящее из n рядов по m участков в каждом. Все участки являются одинаковыми и имеют квадратную форму. Оказалось, что на момент покупки некоторые из этих участков уже удобрены, а некоторые — нет. Редиска растет только на удобренных участках.

Для получения большего урожая Архип решил удобрить некоторый прямоугольный фрагмент поля, состоящий из целых участков. В выбранном фрагменте Архип удобряет каждый участок. Повторное удобрение участка делает его непригодным к выращиванию брюссельской редиски. Закончив удобрять, фермер выбирает для посадки редиски прямоугольный фрагмент поля, состоящий из целых участков, каждый из которых удобрен ровно один раз.

Архип должен выбрать на поле фрагмент для удобрения таким образом, чтобы фрагмент для посадки редиски имел максимальную площадь.

Напишите программу, которая по заданному полю находит фрагмент поля для удобрения и фрагмент поля под посадку.

Формат входных данных

В первой строке входного файла записаны натуральные числа n и m ($2 \leq n \leq 2000$, $2 \leq m \leq 2000$), где n — количество рядов на поле, а m — количество участков в каждом ряду (количество столбцов). Далее в n строках содержится описание поля. Каждая из этих n строк содержит m символов. Символ «1» обозначает, что соответствующий участок поля удобрен, а «0» — не удобрен. Гарантируется, что поле содержит хотя бы один удобренный и хотя бы один неудобренный участок. Поле расположено таким образом, что первая строка его описания соответствует северной стороне, а первый столбец — западной стороне.

Формат выходных данных

Первая строка должна описывать фрагмент поля для удобрения. Фрагмент описывается четырьмя числами a, b, c, d , где a и b — номер ряда и столбца самого северо-западного его участка, а c и d — номер ряда и столбца самого юго-восточного. Ряды нумеруются с севера на юг от 1 до n , а столбцы — с запада на восток от 1 до m .

Вторая строка должна описывать фрагмент под посадку в том же формате.

Третья строка должна содержать площадь фрагмента (количество участков) под посадку.

Если решений несколько, выведите любое.

Примеры входных и выходных данных

farmer.in	farmer.out
4 6	2 3 2 5
111110	1 1 3 4
110010	1 2
111100	
000000	

Система оценивания

Решения, корректно работающие при $n \leq 40$ и $m \leq 40$, будут оцениваться из 30 баллов, а решения, корректно работающие при $n \leq 300$ и $m \leq 300$, будут оцениваться из 60 баллов.