

Задача А. Игра с утюгом

Имя входного файла: `a.in`
 Имя выходного файла: `a.out`
 Максимальное время работы на одном тесте: 2 секунды
 Максимальный объем используемой памяти: 64 мегабайта

Группа московских школьников, участвующих во Всероссийской олимпиаде, сидит за круглым столом и играет в занимательную игру, чтобы расслабиться и отвлечься перед туром. В игру играют с помощью утюга, который в начале игры находится у одного из участников. Этот участник передает утюг своему левому соседу или своему правому соседу с определенной вероятностью. Участник, получивший утюг, проделывает то же самое и т.д. Игра заканчивается, когда утюг побывал у каждого из участников хотя бы один раз. Тот, у кого в этот момент оказался утюг, становится победителем.

Задача состоит в том, чтобы определить, с какой вероятностью победителем станет заданный участник. Вероятности передачи утюга налево или направо индивидуальна для каждого из участников и известна до начала игры.

Участники занумерованы от 1 до N так, что участник номер 2, сидит справа от участника 1, участник 3 — справа от участника 2 и т.д. Участник номер 1 сидит справа от участника номер N . Игра начинается с участника, номер которого указан во входном файле и ваша задача определить вероятность выигрыша участника с номером N .

Формат входных данных

Первая строка входного файла содержит числа N и K ($2 \leq N \leq 50$, $1 \leq K < N$). N задает количество участников, K — номер участника, у которого находится утюг в начале игры.

Вторая строка входного файла содержит $N-1$ чисел, обозначающих вероятность p_i ($0.01 \leq p_i \leq 0.99$) передачи утюга направо для участников, с номерами от 1 до $N-1$. Вероятность передачи утюга налево для участника с номером i равна $1-p_i$. Вероятности даны минимум с двумя знаками после точки.

Формат выходных данных

Выведите одно число, обозначающее вероятность победы участника с номером N с точностью не меньше 6 знаков после точки.

Примеры

<code>a.in</code>	<code>a.out</code>
7 3 0.5 0.5 0.5 0.5 0.5 0.5	0.1666666667
3 1 0.3 0.6	0.3000000000
24 12 0.99 0.99 0.99 0.99 0.99 0.99 0.99 0.99 0.99 0.99 0.99 0.5 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01	0.9800000000

Задача В. Танглеграм

Имя входного файла:

b.in

Имя выходного файла:

b.out

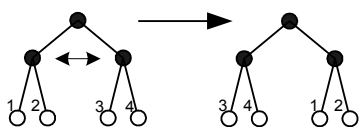
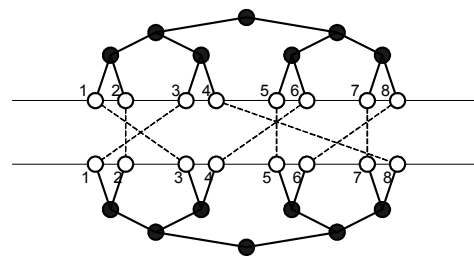
Максимальное время работы на одном тесте:

2 секунды

Максимальный объем используемой памяти:

64 мегабайта

На плоскости расположены два полных бинарных дерева глубины N . Их 2×2^N листьев расположены на двух параллельных прямых и пронумерованы слева направо. Листья с одинаковыми номерами в первом и втором деревьях находятся один напротив другого. Между листьями деревьев задано соответствие – каждый из листьев одного дерева имеет ровно один соответствующий ему лист во втором дереве, и наоборот. На рисунке такие соответствия заданы прямыми отрезками между листьями. Такую конструкцию – два дерева вместе с соответствиями между листьями – называют тангле-грамом. Танглеграммы, например, используются биологами при исследовании взаимосвязей генов разных видов растений.



Танглеграм тяжело исследовать, если многие из отрезков-соответствий пересекаются. Для того, чтобы уменьшить количество пересечений, разрешается осуществлять единственную операцию: в *первом* (верхнем) дереве можно менять местами два поддерева произвольной вершины, как изображено на втором рисунке.

Напишите программу, которая по информации о соответствии между листьями двух деревьев определит наименьшее возможное количество пересечений отрезков в графическом представлении танглеграма, которое может быть достигнуто путем выполнения операций обмена смежных поддеревьев *первого* дерева. Если в одной точке пересекаются больше двух отрезков, то под количеством пересечений нужно понимать количество попарных пересечений отрезков. Например, на первом рисунке соответствия 4-8, 5-5 и 6-4 образуют три пересечения.

Формат входных данных

В первой строке входного файла находится натуральное число N ($1 \leq N \leq 19$) – глубина деревьев. Вторая строка содержит 2^N разных целых чисел от 1 до 2^N , каждое i -ое из которых задает лист второго дерева, который связан отрезком с i -ым листом первого дерева.

Формат выходных данных

Единственная строка выходного файла должна содержать наименьшее возможное количество пересечений отрезков-соответствий в танглеграме, которое может быть достигнуто путем обмена поддеревьев в первом дереве.

Пример

b.in	b.out
3 3 2 1 8 5 4 7 6	6

Сначала, как видно из рисунка, существует 9 пересечений отрезков. Для достижения минимального количества пересечений требуется поменять местами в верхнем дереве поддеревья $(1) \leftrightarrow (2)$, $(5) \leftrightarrow (6)$, $(7) \leftrightarrow (8)$. В данном случае поддеревья состоят из одного листа.

Задача С. Торговец

Имя входного файла: c.in
 Имя выходного файла: c.out
 Максимальное время работы на одном тесте: 2 секунды
 Максимальный объем используемой памяти: 64 мегабайта

Торговец владеет тремя видами товаров: алмазы, яблоки и шелк. Для каждого товара известна стоимость в золотых монетах за единицу веса и его количество у торговца.

В стране, где живет торговец, есть N городов, которые пронумерованы от 1 до N . Родной город торговца имеет номер 1, а столица – номер N . Чтобы добраться до столицы, где торговец может продать товар, ему нужно проехать определенным маршрутом через другие города. Между некоторыми парами городов существуют дороги, проезд по которым стоит определенного количества золотых. В каждом городе взимается налог за провоз каждого из видов товара, заданный в процентах от стоимости провезенного через город товара. Известно, что выехав из любого города, торговец не может в него вернуться. Любые два города соединены не более чем одной дорогой.

Задача торговца получить наибольшую прибыль – разницу полученных в столице денег за проданный товар и расходов на путешествие в столицу. Он не обязан брать с собой весь свой товар. Торговец всегда имеет достаточно золотых для уплаты налогов, и не может рассчитаться товаром, который он везет в столицу. Все дороги ведут только в одном направлении.

Напишите программу, которая по информации о количестве единиц веса разных видов товара у торговца, цене на эти товары в столице, налогах в городах, дорогах между городами и стоимости проезда по этим дорогам установит максимальную прибыль, которую может получить торговец от реализации товара.

Формат входных данных

Первая строка входного файла содержит два целых числа N ($2 \leq N \leq 500$) и M ($M \geq 1$) – количество городов и дорог между ними. Вторая строка содержит три целых неотрицательных числа, которые соответствуют количествам единиц веса алмазов, яблок и шелка, принадлежащим торговцу. Третья строка содержит три целых неотрицательных числа – стоимости единиц веса алмазов, яблок и шелка соответственно. Последующие строки с 4-ой по $N+1$ содержат по три целых числа от 0 до 100 включительно, которые соответствуют процентам от стоимости алмазов, яблок и шелку, которые взимаются, соответственно, в городах от 2 до $N-1$ в качестве налога. В списке городов не учтены родной город торговца 1 и столица N , так как в них с торговца не взимают налог. Последующие M строк содержат по три целых неотрицательных числа, первые два из которых от 1 до N задают пару городов, между которыми проложена дорога, а третье – стоимость проезда по этой дороге. Дороги ведут в направлении от города, указанного первым, до того, который указан вторым. Количества единиц веса каждого из видов товара у торговца, стоимости товаров и цены проезда по дорогам не превышают 100.

Формат выходных данных

Единственная строка выходного файла должна содержать одно число – точное значение найденной максимальной прибыли от поездки в столицу. Ответ всегда должен содержать ровно два знака после точки. В случаях, когда торговец не может получить прибыль или добраться до столицы по существующим дорогам, требуется вывести 0.00

Пример

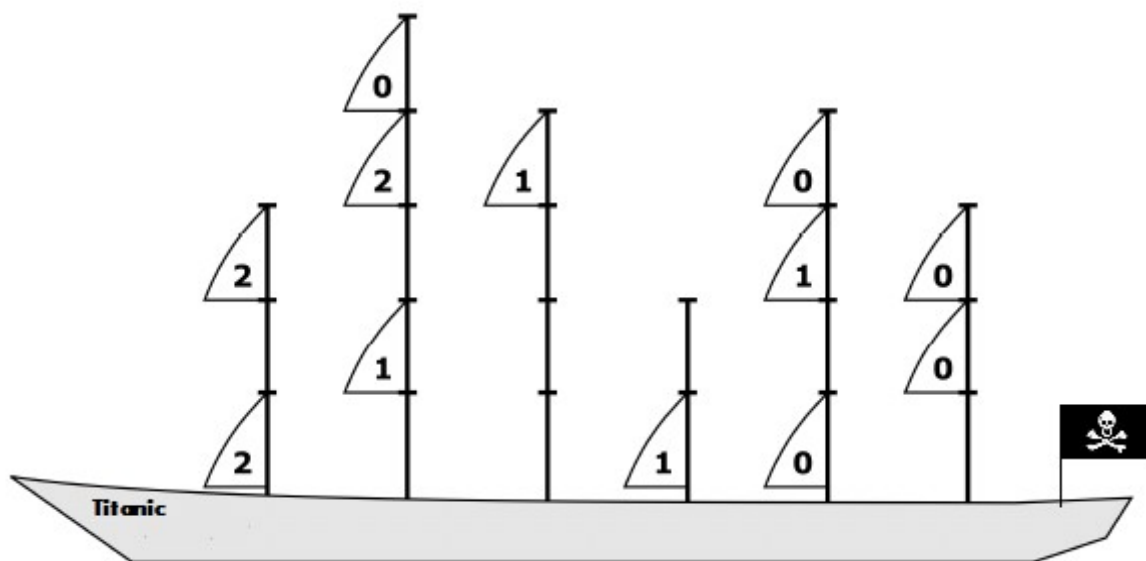
c.in	c.out
4 4 10 5 20 100 5 12 15 40 25 90 20 10 1 2 5 1 3 10 3 4 10 2 4 15	1025.00

Задача D. Паруса

Имя входного файла: d.in
 Имя выходного файла: d.out
 Максимальное время работы на одном тесте: 2 секунды
 Максимальный объем используемой памяти: 64 мегабайта

Сомалийские пираты захватили парусный корабль. На корабле есть N мачт, каждая из которых разбита на сегменты — высота мачты задается количеством сегментов в ней. На каждой мачте находится ряд парусов и каждый парус прикреплен к одному сегменту мачты. Паруса на мачте могут быть произвольно распределены между сегментами, но к каждому сегменту может быть прикреплен только один парус.

Различные конфигурации парусов дают разную тягу, создаваемую ветром. Паруса, находящиеся ближе к носу, чем другие паруса на той же высоте, дают меньшую тягу. Для каждого из парусов определим неэффективность, как количество парусов, находящихся ближе к корме и располагающихся на той же высоте. На рисунке нос находится слева, а корма — справа.



Суммарная эффективность конфигурации равна сумме неэффективностей всех парусов.

В приведенном на рисунке примере, у корабля 6 мачт высотами 3, 5, 4, 2, 4 и 3 считая с носа (слева направо). Показанное распределение парусов дает суммарную неэффективность 10. Неэффективность каждого паруса, написана на нем. Флаг на неэффективность не влияет, он нарисован для красоты.

Напишите программу, которая по данной высоте и количеству парусов на каждой из N мачт, определяет наименьшую суммарную неэффективность.

Формат входных данных

Первая строка входного файла содержит число N ($2 \leq N \leq 100\,000$), количество мачт на корабле.

Каждая из следующих N строк содержит два числа H и K ($1 \leq H \leq 100\,000$, $1 \leq K \leq H$), высоту мачты и количество парусов на ней. Мачты перечислены от носа к корме.

Формат выходных данных

Выведите одно число — наименьшую возможную суммарную неэффективность.

Пример

d.in	d.out
6 3 2 5 3 4 1 2 1 4 3 3 2	10