Задача A. Окружение

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Слева от дома, где живет программист Никанор, строится новый жилой комплекс, а справа от дома Никанора строится новый спортивный комплекс. Каждую из строек от дома Никанора отделяет проезд. К дому Никанора можно попасть только через один из проездов.

Сейчас к обоим строящимся объектам планируют подводить трубы. Этим занимаются разные организации, и каждая из них планирует раскопать для этих целей проезд со «своей» стороны.

Работы по подведению труб к жилому комплексу запланировали начать в день s_1 и вести их в течение d_1 дней. Работы по подведению труб к спортивному комплексу запланировали начать в день s_2 и вести их в течение d_2 дней.

Однако перекрывать оба проезда одновременно даже на один день запрещено, так что эти планы согласовать не удалось. Поэтому каждая из организаций предложила перенести начало работ: те, кто подводит трубы к жилому комплексу, подготовили новый план с переносом начала работ на b_1 дней (т.е. в день $s_1 + b_1$); те же, кто подводит трубы к спортивному комплексу, также подготовили новый план с переносом начала работ на b_2 дней (т.е. в день $s_2 + b_2$). Эти новые планы опять были отправлены на согласование...

Планы отклоняются, если хотя бы в один день будут перекрыты оба проезда. Каждый раз после отклонения очередных планов вносятся два новых плана, в которых начало работ переносится на b_1 дней для жилого комплекса и на b_2 дней для спортивного комплекса.

Ваша задача — определить, в какой день начнутся работы по подведению труб к жилому комплексу и в какой день начнутся работы по подведению труб к спортивному комплексу.

Формат входных данных

В первой строке содержится целое число s_1 ($1 \leqslant s_1 \leqslant 1000$) — день начала работ по подведению труб к жилому комплексу по исходному плану.

Во второй строке содержится целое число d_1 ($1 \leqslant d_1 \leqslant 1000$) — длительность работ по подведению труб к жилому комплексу (в днях).

В третьей строке содержится целое число b_1 ($1 \leqslant b_1 \leqslant 1000$) — количество дней, на которое переносится начало работ по подведению труб к жилому комплексу при каждом отклонении плана.

В четвёртой строке содержится целое число s_2 ($1 \leqslant s_2 \leqslant 1000$) — день начала работ по подведению труб к спортивному комплексу по исходному плану.

В пятой строке содержится целое число d_2 ($1 \leqslant d_2 \leqslant 1000$) — длительность работ по подведению труб к спортивному комплексу (в днях).

В шестой строке содержится целое число b_2 ($1 \leqslant b_2 \leqslant 1000$) — количество дней, на которое переносится начало работ по подведению труб к спортивному комплексу при каждом отклонении плана.

Формат выходных данных

Выведите два целых числа — день, в который начнутся работы по подведению труб к жилому комплексу, и день, в который начнутся работы по подведению труб к спортивному комплексу (когда после всех переносов планы, наконец, будут согласованы). Разделяйте числа пробелом или переносом строки.

Если планы согласовать не получится, выведите в качестве ответа два числа -1, разделённых пробелом или переносом строки.

Система оценки

Во всех подзадачах применяется потестовая система оценки. В графе «Баллы» указано количество баллов за тест и в скобках максимальное количество баллов, которое можно набрать за подзадачу. Участнику сообщаются номера тестов подзадачи, которые не были пройдены.

Для третьей подзадачи требуется, чтобы программа верно решала вторую подзадачу, для пятой подзадачи требуется, чтобы программа верно решала первую, третью и четвертую подзадачи. Более подробно разбиение на подзадачи показано в таблице ниже.

Тесты из условия не оцениваются.

Подзадача	Баллы за тест (баллы за подзадачу)	Ограничения	Необходимые подзадачи	Информация о проверке
1	1 (до 10)	$b_1 = b_2$, все шесть чисел не превосходят 10	нет	полная
2	2 (до 20)	$b_1 = 1$, все шесть чисел не превосходят 10	нет нет	полная полная
3	4 (до 20)	$b_1 = 1$	2	полная
4	2 (до 20)	гарантируется, что планы в какой-то момент будут согласованы	нет	полная
5	2 (до 30)	любые допустимые значения	1, 3, 4	полная

Примеры

стандартный ввод	стандартный вывод
3 4 1 1 5 3	6 10
1 2 5 2 2 5	-1 -1
1 2 4 10 3 1	1 10

Замечание

Поясним приведенные примеры.

Пример 1. Исходные планы: работы у жилого комплекса идут с дня 3 по день 6, у спортивного — с дня 1 по день 5; общие дни —$3, 4, 5$, планы отклонены.

После первого переноса ($b_1 = 1$, $b_2 = 3$) работы у жилого комплекса должны пройти в дни с 4 по 7, у спортивного — в дни с 4 по 8; общие дни — $4, 5, 6, 7$.

После второго переноса предлагается проводить работы у жилого комплекса в дни с 5 по 8, у спортивного — в дни с 7 по 11; общие дни — 7 и 8.

После третьего переноса провести работы у жилого комплекса предлагается в дни с 6 по 9, у спортивного в дни с 10 по 14; общих дней нет, планы согласованы.

Таким образом, ответ — числа 6 и 10, дни начала работ у жилого комплекса и у спортивного комплекса соответственно.

Пример 2. Обе организации предлагают одинаковые переносы ($b_1 = b_2 = 5$), поэтому оба интервала работ всегда сдвигаются на одинаковое число дней: дни 1..2 и 2..3, затем 6..7 и 7..8, затем 11..12 и 12..13 и так далее — общий день есть всегда. Планы согласовать не удастся, поэтому ответ — -1 -1.

Пример 3. Исходные планы состоят в том, чтобы провести работы в дни 1..2 у жилого комплекса и в дни 10..12 у спортивного комплекса; общих дней нет, поэтому планы принимаются сразу, переносов не требуется. Ответ — дни начала работ 1 и 10.

Задача B. Конвейер

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Программист Никанор любит простые в приготовлении блюда. Например, тыквенную запеканку. У Никанора есть очень удобный сотейник, в который помещается m единиц массы тыквы.

Конечно, Никанору было бы удобно, если бы в магазине можно было купить тыквы именно такой массы. Но такое случается редко, обычно Никанор покупает более «весомые» тыквы (и никогда не покупает тыквы массой, меньшей m).

Пусть в первый раз он приобрёл тыкву массой t единиц. Сначала он разрезал её, чтобы очистить изнутри. Затем из m единиц массы он приготовил запеканку. Что касается оставшейся части, то её Никанор тоже поделил: m единиц массы он поместил в холодильник, а остальное — в морозильник. Разумеется, если оставшаяся часть тыквы имеет массу меньше m, то она целиком будет помещена в холодильник.

Часть, хранящаяся в холодильнике, будет полностью израсходована Никанором при приготовлении следующей запеканки; в морозильнике тыква может храниться значительно дольше. К тому же Никанор считает, что в запеканке тыквы из морозильника должно быть не более $m/2$.

Поэтому каждый раз, когда Никанор готовит запеканку, он действует следующим образом.

- Если в холодильнике есть тыква, она будет использована полностью.
- Если масса тыквы из холодильника меньше m, Никанор добавит тыкву из морозильника (если она там есть), но не более $m/2$.
- Если всё ещё не набирается m единиц массы, Никанор использует свежую тыкву. Добавив нужное количество единиц массы для приготовления запеканки, он делит оставшуюся часть свежей тыквы так же, как было описано для самой первой тыквы.

Ваша задача — по заданной последовательности масс тыкв, которые приобретал Никанор, определить, сколько запеканок он приготовил и какое количество тыквы хранится в морозильнике.

Формат входных данных

В первой строке содержится целое число m ($2 \leqslant m \leqslant 1000$) — масса, необходимая для приготовления запеканки. Гарантируется, что число m — чётное.

Во второй строке содержится целое число q ($1 \leqslant q \leqslant 3 \cdot 10^5$) — количество следующих строк.

В каждой из следующих q строк содержится

- либо целое число t ($m \leqslant t \leqslant 1000$) — масса очередной купленной тыквы;
- либо число -1, обозначающее запрос.

Гарантируется, что последнее введённое число равно -1.

Формат выходных данных

В ответ на каждый запрос выведите два целых числа — максимальное суммарное количество запеканок, которые приготовил Никанор из всех приобретённых ранее тыкв, а также суммарную массу тыквы, которая хранится в морозильнике.

Разделяйте числа пробелами или переводами строк.

Ответы выводятся в том же порядке, в котором идут запросы; ответ на каждый запрос существует и является единственным. Если до запроса не было приобретено ни одной тыквы, в качестве ответа следует вывести два нуля.

Система оценки

В первой и второй подзадачах применяется потестовая система оценки. В графе «Баллы» указано количество баллов за тест и в скобках максимальное количество баллов, которое можно набрать за подзадачу. Участнику сообщаются номера тестов подзадачи, которые не были пройдены.

Баллы за третью и четвёртую подзадачи начисляются только в случае прохождения всех тестов этой подзадачи. Участнику сообщается либо номер первого непройденного теста и результат проверки на этом тесте, либо что все тесты подзадачи пройдены.

Для второй и третьей подзадач требуется, чтобы программа верно решала первую подзадачу, для четвёртой подзадачи требуется, чтобы программа верно решала все три предшествующих подзадачи. Более подробно разбиение на подзадачи показано в таблице ниже.

Тесты из условия не оцениваются.

Подзадача	Баллы за тест (баллы за подзадачу)	Ограничения	Необходимые подзадачи	Информация о проверке
1	2 (до 20)	$q \leqslant 100$, $t \leqslant 2m$	нет	полная
2	3 (до 30)	$q \leqslant 100$	1	полная
3	20	$t \leqslant 2m$	1	первая ошибка
4	30	любые допустимые значения	1, 2, 3	первая ошибка

Примеры

стандартный ввод	стандартный вывод
4 6 10 -1 5 -1 4 -1	2 2 3 0 4 0
2 4 7 -1 2 -1	2 3 4 1
4 4 -1 10 6 -1	0 0 4 0
10 5 27 11 -1 10 -1	3 2 4 0

Замечание

Поясним приведённые примеры.

Пример 1. На изготовление запеканки идёт $m = 4$ единицы массы тыквы. Соответственно, из морозильника мы можем брать не более 2 единиц массы на одну запеканку. Продемонстрируем, что происходит, по шагам.

- Тыква массой 10: из 4 единиц готовится запеканка, из оставшихся 6 единиц 4 попадают в холодильник, а 2 — в морозильник. Следующая запеканка будет приготовлена из тыквы, хранящейся в холодильнике; тыква из морозильника пока не может быть использована.
- Поступает первый запрос. Ответом на него будет пара чисел 2 2 — 2 запеканки и 2 единицы массы тыквы в морозильнике.
- В холодильнике тыквы нет, но она есть в морозильнике — как раз в количестве 2 единиц массы, что является максимально допустимым количеством для запеканки. Свежая тыква массой 5 будет разделена на части из 2 единиц массы, которые будут сразу использованы для приготовления запеканки, и 3 единиц массы, которые будут помещены в холодильник ($3 < 4$). Этого количества не хватит для приготовления запеканки, нужна будет новая тыква.
- Поступает второй запрос. Ответом на него будет пара чисел 3 0 — к приготовленным ранее двум добавилась ещё одна запеканка, а в морозильнике тыквы нет.
- Теперь в холодильнике есть 3 единицы массы тыквы, в морозильнике тыквы нет, поэтому 1 единица массы берется из свежей тыквы массой 4. А 3 единицы массы тыквы отправляются в холодильник.
- Поступает третий запрос, ответом на который будет пара чисел 4 0 — приготовлено суммарно 4 запеканки, в морозильнике ничего нет.

Пример 2. На изготовление запеканки идёт 2 единицы массы тыквы, из морозильника можно брать не более 1 единицы массы на одну запеканку.

- Тыква массой 7: из 2 единиц готовится запеканка, оставшиеся 5 единиц делятся на две части: 2 единицы помещаются в холодильник, 3 — в морозильник. Следующая запеканка будет приготовлена из тыквы, хранящейся в холодильнике, а тыква из морозильника пока не может быть использована.
- Поступает первый запрос. Ответом на него будет пара чисел 2 3 — приготовлено 2 запеканки, в морозильнике 3 единицы тыквы.
- В холодильнике тыквы нет, для использования тыквы из морозильника нужна свежая тыква. Приобретённая тыква массой 2 будет поделена на две равные части: одна пойдёт на приготовление запеканки вместе с 1 единицей массы тыквы из морозильника, а ещё одна будет помещена в холодильник. Следующая запеканка будет приготовлена из 1 единицы массы тыквы из холодильника и 1 единицы массы тыквы из морозильника. После этого в морозильнике останется 1 единица массы, которая не может быть использована без свежей тыквы.
- Поступает второй запрос. Ответом на него будет пара чисел 4 1 — было приготовлено 4 запеканки и 1 единица массы тыквы остаётся в морозильнике.

Пример 3. На изготовление запеканки идёт 4 единицы массы тыквы, из морозильника можно брать не более 2 единиц массы на одну запеканку.

- Поступает первый запрос. Поскольку ещё ни одной тыквы не было приобретено, ответом на него будет пара чисел 0 0 — не приготовлено ни одной запеканки, морозильник пуст.
- Тыква массой 10: из 4 единиц готовится запеканка, из оставшихся 6 единиц 4 помещаются в холодильник, 2 — в морозильник. Следующая запеканка будет приготовлена из тыквы, хранящейся в холодильнике. В морозильнике останется 2 единицы массы тыквы, которая может быть использована только вместе со свежей тыквой.

- Тыква массой 6: холодильник пуст, из морозильника берутся 2 единицы массы тыквы, и ещё 2 — из свежей тыквы, и готовится третья запеканка. Оставшиеся 4 единицы массы тыквы помещаются в холодильник. Следующая запеканка полностью готовится из тыквы, хранящейся в холодильнике.

- Поступает второй запрос, ответом на который будет пара чисел 4 0: приготовлено 4 запеканки, морозильник пуст.

Пример 4. На изготовление запеканки идёт 10 единиц массы тыквы, из морозильника можно брать не более 5 единиц массы на одну запеканку.

- Тыква массой 27: из 10 единиц готовится запеканка, из оставшихся 17 единиц 10 помещаются в холодильник, 7 — в морозильник. Следующая запеканка будет приготовлена из тыквы, хранящейся в холодильнике. В морозильнике останется 7 единиц массы тыквы, которая может быть использована только вместе со свежей тыквой.

- Тыква массой 11: холодильник пуст, из морозильника берутся 5 единиц массы тыквы (там остаётся 2 единицы), ещё 5 — из свежей тыквы, и готовится третья запеканка. Оставшиеся 6 единиц массы тыквы помещаются в холодильник ($6 < 10$). Вместе с 2 единицами из морозильника получается только 8 единиц массы, этого не хватит для приготовления запеканки, нужна будет новая тыква.

- Поступает первый запрос. Ответом на него будет пара чисел 3 2 — приготовлено 3 запеканки, в морозильнике 2 единицы массы тыквы.

- Тыква массой 10: в запеканку идут все 6 единиц массы тыквы из холодильника, все 2 единицы из морозильника и ещё 2 единицы из свежей тыквы — готовится четвёртая запеканка. Оставшиеся 8 единиц массы тыквы помещаются в холодильник, морозильник пуст.

- Поступает второй запрос, ответом на который будет пара чисел 4 0: приготовлено 4 запеканки, морозильник пуст.

Задача C. Фишки

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В компании, в которой работает Никанор, устроили развлекательные соревнования для сотрудников. В каждом соревновании участник зарабатывает и тратит фишки.

Сейчас по плану организаторов сотрудники должны разделиться на пары и сыграть между собой, расходуя уже накопленные фишки. Мы не будем описывать правила игры, но скажем, что если количество фишек у игроков в паре сильно отличается, играть им будет менее интересно.

Поскольку игроки уже разделились на пары (и не хотят менять это разделение), организаторы объявили, что они приготовили коробочки с сюрпризом по количеству пар игроков. При этом в некоторых коробочках лежат фишки (одинаковое количество в каждой), а в некоторых — пусто. Коробочку выдадут тому игроку из пары, у которого фишек меньше (при равенстве выберут получателя случайным образом).

Конечно, задумка организаторов состоит в том, чтобы выдать коробочки с фишками тем участникам, которые имеют большую разницу в количестве фишек со своими напарниками.

Ваша задача — определить, сколько фишек организаторы должны положить в непустые коробочки, чтобы максимальная разница в количестве фишек между игроками стала минимально возможной. Также нужно будет указать, какие пары получат коробочки с фишками.

Формат входных данных

В первой строке содержится целое число n ($2 \leqslant n \leqslant 10^5$) — количество пар игроков.

В каждой из следующих n строк содержится по два целых числа f_j и s_j через пробел ($1 \leqslant f_j, s_j \leqslant 10^9$, $j = 1, 2, \ldots, n$) — количество фишек у первого и у второго игрока в j-й паре соответственно.

Формат выходных данных

Выведите два целых числа b и d ($0 \leqslant b, d \leqslant 10^9$), а также строку S длины n, состоящую из цифр 0 и 1.

Число b — количество фишек, которое организаторам следует положить в непустые коробочки.

Число d — минимально возможная максимальная разница в итоговом количестве фишек между игроками.

Строка S описывает распределение коробочек: 0 на позиции k означает, что пара $\#k$ получит пустую коробочку, а 1 — что пара $\#k$ получит коробочку с b фишками.

Если существует несколько правильных ответов, выведите любой из них.

Система оценки

В первой подзадаче применяется потестовая система оценки. В графе «Баллы» указано количество баллов за тест и в скобках максимальное количество баллов, которое можно набрать за подзадачу. Участнику сообщаются номера тестов подзадачи, которые не были пройдены.

Баллы за вторую, третью и четвёртую подзадачи начисляются только в случае прохождения всех тестов этой подзадачи. Участнику сообщается либо номер первого непройденного теста и результат проверки на этом тесте, либо что все тесты подзадачи пройдены.

Для второй и третьей подзадач требуется, чтобы программа верно решала первую подзадачу, для четвёртой подзадачи требуется, чтобы программа верно решала все три предшествующих подзадачи. Более подробно разбиение на подзадачи показано в таблице ниже.

Тесты из условия не оцениваются.

Подзадача	Баллы за тест (баллы за подзадачу)	Ограничения	Необходимые подзадачи	Информация о проверке
1	2 (до 20)	$n \leqslant 10$, $f_j, s_j \leqslant 1000$	нет	полная
2	25	$n \leqslant 1000$	1	первая ошибка
3	25	$\lvert f_j - s_j \rvert \leqslant 1000$ для каждой пары	1	первая ошибка
4	30	любые допустимые значения	1, 2, 3	первая ошибка

Примеры

стандартный ввод	стандартный вывод
3 1 10 5 5 3 7	6 3 101
2 1 5 10 2	6 2 11
2 7 7 2 2	0 0 00

Замечание

Поясним приведённые примеры.

Пример 1. Разницы в парах: $|1 - 10| = 9$, $|5 - 5| = 0$, $|3 - 7| = 4$.

При $b = 6$ коробочки с фишками получают первая и третья пары:

- первая пара: игрок с 1 фишкой получает 6, разница становится $|7 - 10| = 3$;
- вторая пара получает пустую коробочку, её разница остаётся прежней: 0 (коробочка с 6 фишками только увеличила бы разницу в этой паре);
- третья пара: разница становится $|9 - 7| = 2$.

Максимальная разница в количестве фишек между игроками составляет 3 (в первой паре).

Подошёл бы и ответ $b = 7$ с тем же распределением коробочек с фишками. В этом случае разницы в количестве фишек в парах были бы 2, 0 и 3 соответственно, а максимальная разница снова составила бы 3 (в третьей паре).

Разницу меньше 3 получить нельзя.

Пример 2. Разницы в парах равны 4 и 8. При $b = 6$ обе пары получают коробочки с фишками, и разницы становятся $|7 - 5| = 2$ и $|10 - 8| = 2$.

Меньшей разницы добиться не получится: одно и то же число фишек b должно одновременно подойти и паре с разницей 4, и паре с разницей 8.

Пример 3. В каждой паре у игроков уже одинаковое количество фишек, поэтому ни одной паре не следует выдавать коробочку с фишками. При таком распределении число b может быть любым допустимым — в примере это 0.

Задача D. Благоустройство

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Никанор любит ходить пешком на работу. Его путь пролегает по улице, вдоль которой стоят большие деревья. Никанору нравится там ходить: летом листва защищает от палящего солнца, зимой заснеженные ветви образуют причудливые узоры...

Впрочем, городские власти имеют другое мнение на этот счёт: они полагают, что деревьям в городе не место, и планируют срубить все n деревьев. А чтобы возмущённые жители не могли воспрепятствовать вырубке, власти планируют заказать экспертизу, которая признает, что деревья были заражены вредителем.

Для каждого дерева известна стоимость акта экспертизы f_j. При этом, если дерево $\#j$ признаётся заражённым, то стоимость акта экспертизы для деревьев, являющихся его непосредственными соседями слева (дерево $\#(j-1)$) и справа (дерево $\#(j+1)$), уменьшается вдвое. Поэтому власти планируют заказать экспертизу в два этапа: сначала признать некоторые деревья заражёнными, а затем заказать экспертизу для их непосредственных соседей (если таковые имеются). Разумеется, цель состоит в том, чтобы потратить на экспертизу как можно меньше денег.

Ваша задача — определить минимальное количество денег, за которое может быть проведена экспертиза всех деревьев, а также номера деревьев, для которых экспертиза будет заказана на первом этапе.

Формат входных данных

В первой строке содержится целое число n ($1 \leqslant n \leqslant 2 \cdot 10^5$) — количество деревьев.

Во второй строке содержится n целых чётных чисел $f_1, f_2, \ldots, f_n$ ($2 \leqslant f_j \leqslant 10^9$), где f_j — стоимость акта экспертизы дерева $\#j$, если она будет заказана на первом этапе.

Формат выходных данных

В первой строке выведите целое число s — минимальную суммарную стоимость экспертизы всех деревьев.

Во второй строке выведите целое число m — количество деревьев, для которых экспертиза будет выполнена на первом этапе.

Далее выведите m целых чисел — номера деревьев, для которых экспертиза будет выполнена на первом этапе, в порядке возрастания. Разделяйте числа пробелом или переводом строки.

Если существует несколько вариантов набора деревьев, для которых экспертиза выполняется на первом этапе, выведите любой из них.

Важно: не требуется, чтобы число m было минимально возможным.

Система оценки

В первой подзадаче применяется потестовая система оценки. В графе «Баллы» указано количество баллов за тест и в скобках максимальное количество баллов, которое можно набрать за подзадачу. Участнику сообщаются номера тестов подзадачи, которые не были пройдены.

Баллы за подзадачи со второй по пятую начисляются только в случае прохождения всех тестов этой подзадачи. Участнику сообщается либо номер первого непройденного теста и результат проверки на этом тесте, либо что все тесты подзадачи пройдены.

Для второй подзадачи требуется, чтобы программа верно решала первую подзадачу, для пятой подзадачи требуется, чтобы программа верно решала все четыре предшествующих подзадачи. Более подробно разбиение на подзадачи показано в таблице ниже.

Тесты из условия не оцениваются.

Подзадача	Баллы за тест (баллы за подзадачу)	Ограничения	Необходимые подзадачи	Информация о проверке
1	1 (до 20)	$n \leqslant 15$, $f_j \leqslant 1000$	нет	полная
2	20	$n \leqslant 1000$, $f_j \leqslant 1000$	1	первая ошибка
3	20	$f_j \leqslant 4$	нет	первая ошибка
4	20	$f_j = 2 \cdot j$ для всех j	нет	первая ошибка
5	20	любые допустимые значения	1, 2, 3, 4	первая ошибка

Примеры

стандартный ввод	стандартный вывод
4 10 2 4 8	15 2 2 3
1 6	6 1 1
5 2 2 2 2 2	7 2 2 4

Замечание

Поясним приведённые примеры.

Пример 1.

На первом этапе заражёнными признаются деревья #2 и #3: за акты экспертизы для них платится полная стоимость $2 + 4 = 6$.

На втором этапе экспертиза проводится для деревьев #1 и #4:

- дерево #1 — соседнее по отношению к дереву #2, признанному заражённым на первом этапе, поэтому акт экспертизы для него обойдётся в $10/2 = 5$;
- дерево #4 — соседнее по отношению к дереву #3, признанному заражённым на первом этапе, поэтому акт экспертизы для него обойдётся в $8/2 = 4$.

Итоговая сумма составит $6 + 5 + 4 = 15$.

Покажем, что получить меньшую сумму невозможно.

Полную стоимость экспертизы придется заплатить не менее чем для двух деревьев: выбор любого одного дерева даст возможность снизить стоимость экспертизы только для его непосредственных соседей (или соседа), и как минимум одно из деревьев не будет иметь соседа с актом экспертизы.

Выбор любой другой пары деревьев (кроме 2 и 3) для первого этапа, позволяющий провести второй этап для двух оставшихся деревьев как для соседей заражённых, обойдётся дороже: для пары $\{2, 4\}$ — 17, для пары $\{1, 3\}$ — 19, для пары $\{1, 4\}$ — 21.

Пример 2. У единственного дерева нет соседей, поэтому его экспертизу можно заказать только на первом этапе и только за полную стоимость 6.

Пример 3. Стоимости всех актов экспертизы одинаковы, поэтому выгодно на первом этапе признать заражёнными как можно меньше деревьев (но так, чтобы все остальные считались соседями заражённых).

Одно дерево имеет не более двух соседей, поэтому на первом этапе следует провести экспертизу деревьев #2 и #4: дерево #2 соседнее для деревьев #1 и #3, дерево #4 — соседнее для деревьев #3 и #5.

Итоговая сумма 7 складывается из $2+2 = 4$ на первом этапе и $3 \cdot 1 = 3$ за три акта по половинной цене на втором этапе.