

Задача А. Электронный счетчик

Правительство Байтландии приняло решение о поддержке отечественных производителей электронных компонент. Одна из компаний разработала жидкокристаллические дисплеи для электронных счетчиков, которые могут отображать цифры. Каждая цифра изображается с помощью семи сегментов, каждый из которых может быть включен или выключен (вы наверняка видели подобные на электронных часах). Изображение цифр приведено на рисунке:



Все счетчики будут вести отсчет от нуля (он отображается на экране в начале работы) до некоторого числа k , последовательно проходя по всем значениям. Ведущие нули в числах, отображаемых на счетчиках, не отображаются, все сегменты перед первой значащей цифрой выключены.

За одну операцию можно изменить состояние одного сегмента (включить или выключить). Компания получила заказ на изготовление n типов счетчиков, для каждого из типов известно максимальное значение счетчика k_i . Для каждого из типов счетчика определите, какое количество операций по изменению состояния сегмента необходимо будет произвести для прохода по всем значениям от нуля до k_i .

В первой строке входных данных записано число t — количество различных типов счетчиков. В следующих t строках записаны числа k_i . Ответ для каждого типа счетчика выведите в отдельной строке.

В первом тесте $t = 3$, k_i не превосходит 100. Оценка за этот тест: 30 баллов. За каждое правильно подсчитанное число операций начисляется 10 баллов. Проверка осуществляется в режиме online (результат виден сразу).

Во втором тесте $t = 700$, k_i не превосходит 10^9 . Оценка за этот тест: 70 баллов. За каждое правильно подсчитанное число операций начисляется 0.1 балла. Во время тура проверяется, что сданный файл содержит 700 чисел. Проверка правильности ответа осуществляется в режиме offline (результат виден после окончания тура).

Примеры

Входные данные	Результат
3	9
2	32
10	11
3	

Задача В. Эффективные скидки

В связи с тяжелой экономической ситуацией байтландский интернет-магазин стал получать меньше прибыли. Однако эффективные менеджеры нашли неожиданный способ увеличить доходы — предоставлять клиентам скидки!

Для каждого заказа известна стоимость купленных товаров, стоимость доставки, а также сумма, начиная с которой доставка бесплатна (то есть если стоимость купленных товаров больше либо равна этой сумме, то доставка не оплачивается). Все стоимости выражаются целым количеством байтландских бурлей.

На товары в заказе можно предоставить скидку, скидка также выражается целым количеством байтландских бурлей.

Для каждого заказа определите максимальную суммарную стоимость купленных товаров и стоимость доставки, с учетом возможности предоставления скидки.

В первой строке входных данных записано число t — количество различных заказов. В следующих t строках записаны по три числа $cost$ — стоимость купленных товаров, $deliverycost$ — стоимость доставки и $freedelivery$ — стоимость купленных товаров, начиная с которой доставка бесплатна. Ответ для каждого заказа выведите в отдельной строке.

В первом тесте $t = 3$. Оценка за этот тест: 30 баллов. За каждую правильно подсчитанную максимальную стоимость начисляется 10 баллов. Проверка осуществляется в режиме online (результат виден сразу).

Во втором тесте $t = 700$. Оценка за этот тест: 70 баллов. За каждую правильно подсчитанную максимальную стоимость начисляется 0.1 балла. Во время тура проверяется, что сданный файл содержит 700 чисел. Проверка правильности ответа осуществляется в режиме offline (результат виден после окончания тура).

Примеры

Входные данные	Результат
4	1000
1000 100 10	1000
1000 100 100	1099
1000 100 1000	1100
1000 100 10000	

Задача С. Черно-белый экран

Электронная промышленность Байтландии развивается и одна из компаний начала выпускать черно-белые квадратные экраны размером $n \times n$ пикселей.

Для экономии траффика был разработан специальный режим экрана, в котором при вызове функции с параметрами row, col изменяется состояние всех пикселей в строке row и столбце col (то есть черные пиксели становятся белыми, а белые — черными). Пиксел на пересечении строки row и col также меняет свое состояние.

Чтобы продемонстрировать возможности экономии траффика необходимо разработать последовательность вызовов функции для формирования определенного изображения. Изначально весь экран черный (все пиксели находятся в состоянии 0).

В первой строке входных данных записано число t — количество различных изображений, которые необходимо сформировать на экране. Затем описывается t блоков. Описание блока содержит число n — размер экрана, а затем n строк по n чисел 0 или 1 в каждой — изображение, которое необходимо сформировать на экране.

В качестве ответа необходимо сформировать t блоков с параметрами вызова функции для формирования изображения на экране. Описание каждого блока должно состоять из числа k — количества вызовов функции для генерации изображения и k пар чисел row, col , задающих параметры функции. Нумерация начинается с левого верхнего угла, с нуля.

Формула оценки за каждый блок, при условии, что в результате вызовов функции получается требуемое изображение: $5 + 5 \times (BestAns/PartAns)^4$, где $PartAns$ — количество вызовов функции в решении участника, а $BestAns$ — минимальное количество вызовов функции среди решений всех участников и жюри.

В первом тесте $t = 3$. Оценка за этот тест: 30 баллов. Проверка осуществляется в режиме online (результат виден сразу).

Во втором тесте $t = 7$. Оценка за этот тест: 70 баллов. Во время тура проверяется, что сданный файл содержит корректное описание t блоков (не обязательно формирующих правильное изображение). Проверка правильности ответа осуществляется в режиме offline (результат виден после окончания тура).

Примеры

Входные данные	Результат
3	1
4	2 2
0 0 1 0	2
0 0 1 0	2 2
1 1 1 1	1 1
0 0 1 0	2
4	2 2
0 1 1 0	0 0
1 1 0 1	
1 0 1 1	
0 1 1 0	
3	
1 1 0	
1 0 1	
0 1 1	