

Задача 1. Доблесть и честь

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

На этот раз царю Леониду предстоит защитить Спарту от врагов. В распоряжении у него имеется N бойцов. i -й боец обладает верностью A_i единиц. Этим бойцам предстоит отразить M вражеских отрядов. Сила i -го отряда равна B_i единиц. Полк бойцов отражает атаку i -го вражеского отряда, если каждый боец полка имеет верность строго больше B_i и количество бойцов в полке не меньше количества бойцов во вражеском отряде (**количество бойцов во вражеском отряде равно его силе**). Каждый спартанец полка при столкновении с вражеским отрядом погибает смертью храбрых. Леонид хочет знать, может ли он сформировать несколько полков из спартанцев так, чтобы отбить атаки всех вражеских отрядов. (Замечание: один полк отбивает атаку только одного вражеского отряда)

Формат входных данных

В первой строке записано число T ($1 \leq T \leq 100$) — количество тестов. После этого следует описание каждого тестового случая. В первой строке описания содержится целое число N ($1 \leq N \leq 10^5$) — количество спартанцев. Во второй строке содержится массив A из N целых чисел ($1 \leq A_i \leq 10^9$) — верность каждого спартанца. В третьей строке содержится целое число M ($1 \leq M \leq 10^5$) — количество вражеских отрядов. В четвертой строке содержится массив B из M целых чисел ($1 \leq B_i \leq 10^9$) — силу каждого отряда. Гарантируется, что сумма N и M по всем входным данным не превышает 10^5 .

Формат выходных данных

В ответ выведите T строк — ответ для каждого тестового случая: может ли Леонид собрать несколько полков. Если может, то выводите «Yes» (без кавычек), иначе «No».

Пример

стандартный ввод	стандартный вывод
3	Yes
5	Yes
4 6 9 2 1	No
2	
1 3	
5	
3 8 5 10 4	
2	
3 1	
5	
9 7 9 6 9	
2	
5 4	

Задача 2. Прямоугольные треугольники

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

На координатной плоскости заданы N точек с целочисленными координатами.

На некоторых тройках этих точек можно построить прямоугольные треугольники. Напишите программу, которая найдет все такие треугольники и посчитает их количество.

Формат входных данных

Первая строка входных данных содержит число N ($1 \leq N \leq 100$) — количество точек.

Каждая из следующих N строк содержит пару чисел x_k, y_k ($-10^9 \leq x_k, y_k \leq 10^9$) — координаты точек, разделённые пробелом.

Гарантируется, что все точки попарно различны.

Формат выходных данных

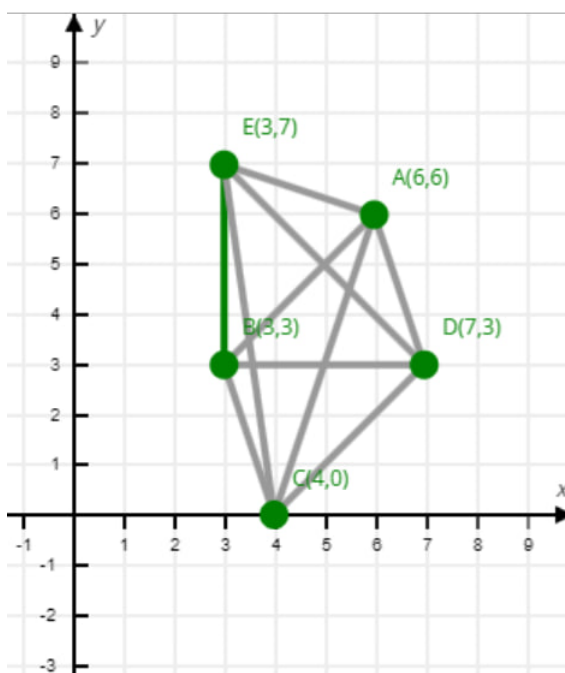
Выведите одно число, искомое количество прямоугольных треугольников.

Пример

стандартный ввод	стандартный вывод
6 10 1 3 3 6 6 3 7 7 3 4 0	3

Замечание

На рисунке ниже прямоугольными треугольниками являются треугольники ACE , CDE и BDE .



Задача 4. Даня и тульские пряники

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

Студент Даня получил стипендию, и на неё купил себе T тульских пряников.

Тульский пряник можно представить, как клетчатый прямоугольник, где у каждой клетки есть уровень вкусоности, представленный целым числом.

Так как Даня не любит прямоугольники, а любит квадраты, то он хочет вырезать из каждого своего пряника кусочек квадратной формы, у которого сумма вкусоностей всех попавших в него клеток максимальна, а затем съесть вырезанные кусочки.

Формат входных данных

В первой строке содержится целое число T ($1 \leq T \leq 50$) — количество купленных тульских пряников.

Каждый тульский пряник описывается в первой строке двумя числами N и M ($1 \leq N, M \leq 500$) — высотой и шириной тульского пряника соответственно.

Затем следует N строк по M чисел, по модулю не превышающих $5 \cdot 10^4$ (но возможно отрицательных), которые описывают вкусоность каждого кусочка пряника.

Гарантируется, что у каждого пряника есть хотя бы одна клетка, у которой неотрицательная вкусоность. Сумма высот тульских пряников не превосходит 500.

Формат выходных данных

Для каждого тульского пряника в отдельной строке выведите максимальную сумму вкусоностей по всем кусочкам квадратной формы.

Примеры

стандартный ввод	стандартный вывод
<pre> 1 4 3 -6 2 1 -3 -5 -4 -6 7 -6 5 -9 -10 </pre>	<pre> 7 </pre>
<pre> 3 2 5 1 2 -1 -3 5 -4 -1 -2 1 5 3 4 -3 8 6 4 -8 -7 -5 3 -5 -5 8 6 2 2 -5 5 4 -6 </pre>	<pre> 8 18 5 </pre>

Задача 5. Жизнь пешки

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

Компьютер записывает «биографию» белой шахматной пешки в течение партии на доске 8×8 . Начальное положение пешки — клетка E2.

На каждом ходе пешка может остаться на месте, сдвинуться вперед на одну клетку или на две клетки, если это ее первый ход.

Компьютер в начале игры и после каждого хода белых записывает позицию пешки по вертикали. Если пешка была «взята», запись позиции прекращается. Считать, что пешку могут взять сразу после первого хода. При этом пешка не может двигаться вперед с позиции E8, но может стоять на этой позиции сколько угодно ходов до конца партии.

Вам дана длина шахматной партии, найдите количество всех возможных «жизненных путей» пешки по модулю 998244353.

Формат входных данных

Единственная строка входных данных содержит целое число N ($1 \leq N \leq 10^6$) — длину партии.

Формат выходных данных

В ответе выведите искомое количество различных шахматных путей по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
2	10
99	582018135

Замечание

10 возможных путей, если партия длилась два хода: 22, 23, 24, 222, 223, 224, 233, 234, 244, 245.

Задача 6. Владелец банка

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

Бизнесмен Владимир является владельцем самого крупного банка в мире «Банк-Вованк». Чтобы подчеркнуть свой статус, он каждый день приходит в автосалон с целью прикупить новых машин в свою коллекцию.

В автосалоне есть N различных марок машин. Количество машин i -й марки равно A_i , каждая из них имеет цену B_i бурлей. Владимир будет приходить в автосалон в течение Q дней. В i -й день он хочет купить C_i самых дорогих машин, стоимость каждой из которых не превышает D_i бурлей (при этом не обязательно все машины должны иметь различную марку). При этом при равенстве в цене он отдаёт приоритет машинам, у которых номер марки меньше (т.е. той марке, которая раньше записана во входных данных). Если нашлось столько машин с подходящей стоимостью, Владимир купит все эти машины. Однако если столько машин не нашлось, он не будет покупать ничего.

Формат входных данных

Первая строка содержит число N ($1 \leq N \leq 10^5$) — количество марок машин в автосалоне. Вторая строка содержит массив A из N чисел ($1 \leq A_i \leq 10^9$) — количество машин каждой марки. Третья строка содержит массив B из N чисел ($1 \leq B_i \leq 10^9$) — стоимость одной машины каждой марки. Четвёртая строка содержит число Q ($1 \leq Q \leq 10^5$) — в течение сколько дней Владимир будет приходить в автосалон. Пятая строка содержит массив C из Q чисел ($1 \leq C_i \leq 10^9$) — количество машин, которое он хочет купить каждый день. Шестая строка содержит массив D из Q чисел ($1 \leq D_i \leq 10^9$) — ограничение на стоимость машин каждый день.

Формат выходных данных

Для каждого дня в отдельной строке выведите «Yes», если Владимир совершил покупку, либо «No», если не совершил (выводите без кавычек). В последней строке выведите N чисел — количество оставшихся машин каждой марки (в том же порядке, что и во входных данных).

Система оценки

Решения, правильно работающие для $N \cdot Q \leq 10^6$, будут набирать не менее 50 баллов.

Пример

стандартный ввод	стандартный вывод
5	Yes
4 6 9 2 1	Yes
1 8 3 8 5	No
3	4 2 0 2 0
10 4 8	
6 9 7	