

## Задача А. Рюкзак

Имя входного файла: `knapsack.in`  
Имя выходного файла: `knapsack.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 8 мегабайт

Найдите максимальную цену слитков золота, которые можно унести в рюкзаке вместительностью  $S$ , если есть  $N$  золотых слитков с заданными весами и ценами.

### Формат входных данных

В первой строке входного файла записаны два числа —  $S$  и  $N$  ( $1 \leq S \leq 10\,000$ ,  $1 \leq N \leq 300$ ).

В двух следующих строках записано по  $N$  неотрицательных целых чисел в каждой — веса и стоимости слитков, соответственно. Каждое из этих чисел не превосходит 100 000.

### Формат выходных данных

Выведите искомую максимальную цену

### Примеры

<code>knapsack.in</code>	<code>knapsack.out</code>
10 3	123
1 4 8	
72 7 51	

## Задача В. Покупка билетов

Имя входного файла: `tickets.in`  
Имя выходного файла: `tickets.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

За билетами на премьеру нового мюзикла выстроилась очередь из  $N$  человек, каждый из которых хочет купить 1 билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более 3-х билетов в одни руки, поэтому договориться таким образом между собой могли лишь 2 или 3 подряд стоящих человека.

Известно, что на продажу  $i$ -му человеку из очереди одного билета кассир тратит  $A_i$  секунд, на продажу двух билетов —  $B_i$  секунд, трех билетов —  $C_i$  секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

### Формат входных данных

Во входном файле записано сначала число  $N$  — количество покупателей в очереди ( $1 \leq N \leq 5000$ ). Далее идет  $N$  троек натуральных чисел  $A_i, B_i, C_i$ . Каждое из этих чисел не превышает 3600. Люди в очереди нумеруются начиная от кассы.

### Формат выходных данных

В выходной файл выведите одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

### Примеры

<code>tickets.in</code>	<code>tickets.out</code>
5	12
5 10 15	
2 10 15	
5 5 5	
20 20 1	
20 1 1	

## Задача С. Футболки

Имя входного файла: `tshirts.in`  
Имя выходного файла: `tshirts.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

На зарядку сегодня утром пришло  $N$  ЛКШат, они построились в ряд. Разумеется, ребята ходят в разноцветных футболках. Никита Павлович, наш многоуважаемый физрук, заметил, что можно попросить некоторых ребят присесть, и тогда для ребят, которые останутся стоять, будет выполнено следующее: последовательность цветов их футболок при перечислении слева направо будет такой же как и последовательность при перечислении справа налево, то есть будет *палиндромом*.

Например, если на зарядку пришли Ксюша в зеленой футболке, Слава в желтой, Артём в красной и Вася в зеленой, то можно попросить присесть Славу, тогда последовательность цветов будет «зеленый, красный, зеленый» как слева направо, так и слева направо. Аналогично можно попросить присесть Артёма (последовательность будет «зеленый, желтый, зеленый»), Славу и Артёма одновременно или любой набор из троих ребят. Таким образом, всего есть 7 способов добиться того, чтобы последовательность цветов была палиндромом.

Помогите Никите Павловичу найти количество способов попросить некоторых ЛКШат присесть, чтобы последовательность цветов футболок оставшихся стоять была палиндромом. Поскольку это число может быть очень большим, выведите его по модулю  $10^9$ .

### Формат входных данных

Первая строка входного файла содержит число  $N$  — количество ЛКШат, пришедших на зарядку ( $1 \leq N \leq 2000$ ). Вторая строка содержит  $N$  целых чисел, каждое из которых задает цвет футболки ЛКШонка и изменяется в пределах от 1 до  $10^9$ . Разные цвета задаются разными числами, а одинаковые — одинаковыми.

### Формат выходных данных

Выведите в выходной файл одно число — искомое количество способов по модулю  $10^9$ .

### Примеры

tshirts.in	tshirts.out
4 1 2 3 1	7

### Задача D. Горец

Имя входного файла: highlander.in  
Имя выходного файла: highlander.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Уже  $n$  дней в ЛКШ идет игра горец, правила которой вам хорошо знакомы. Культурги Даша и Катя, наблюдающие за игрой, каждый день обсуждают количество людей, которые уже успел убить их любимый школьник Никита. Даша говорит, что в день  $i$  никита убил  $a_i$  человек, а Катя утверждает, что он убил  $b_i$  человек. Известно, что Даша может несколько приуменьшить заслуги Никиты, и сказать число меньшее, чем количество убитых им человек, а Катя наоборот склонна присваивать Никите большее количество жертв. Определите количество способов распределения реального количества жертв по дням, если известно, что в сумме за  $n$  дней Катя и Даша ошиблись на одинаковое количество человек.

### Формат входных данных

В первой строке входного файла содержится натуральное число  $n$  ( $1 \leq n \leq 20$ ). В следующих  $n$  строчках через пробел записаны пары  $(a_i, b_i)$ . При этом  $0 \leq a_i \leq b_i \leq 100$ .

### Формат выходных данных

Выходной файл должен содержать число способов различных вариантов распределения жертв по дням по модулю  $10^9 + 7$ . Если ни одного такого способа не существует, то ответ — 0.

### Примеры

highlander.in	highlander.out
2 1 2 2 5	2

### Задача E. Помехи при передаче информации

Имя входного файла: interference.in  
Имя выходного файла: interference.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

При передаче информации в Битландии используется побитовый канал для связи, то есть информация передается по одному биту. Однако, стабильности в такой передаче достигнуть не удалось, и на канале происходят помехи. Поэтому передаваемые двоичные цифры могут заменяться (вместо ноля приходит единица, и наоборот), пропускаться или случайно дублироваться.

После передачи очередного важного сообщения, два оператора (передающий информацию, и получающий) решили встретиться и обсудить процесс передачи информации. А именно они хотят понять, насколько все плохо, то есть какое минимальное количество помех произошло при передаче сигнала. Каждый пропуск, дублирование и замена считаются одной помехой. Помогите им посчитать это количество.

### Формат входных данных

В первой строке дано одно число  $n$  — количество цифр в исходном сообщении. Во второй строке содержится исходное сообщение, состоящее из нулей и единиц. В третьей строке дано одно число  $m$  — количество цифр в полученном сообщении. В четвертой строке содержится полученное сообщение, состоящее из нулей и единиц ( $1 \leq n, m \leq 5000$ ).

### Формат выходных данных

Выведите одно число — минимальное количество помех.

### Примеры

interference.in	interference.out
6 101100 6 110111	3
10 0111010110 12 010000001010	5