**Фафа и его компания**

Фафа руководит компанией, которая работает над большими проектами. Всего в компании Фафы *n* работников. Когда появляется новый проект, Фафа должен распределить задачи по проекту между всеми работниками.

Фафа недавно понял, что каждый раз делать это очень сложно. Поэтому он решил выбрать лучших *l* сотрудников из компании в руководители. Когда появляется новый проект, Фафа будет распределять задачи только между руководителями, а каждый руководитель будет отвечать за некоторое положительное число сотрудников и будет распределять задачи между ними. Чтобы все было честно, каждый руководитель должен получить одинаковое число сотрудников под свое руководство. Кроме того, каждый сотрудник, не являющийся руководителем, должен быть подчинен ровно одному руководителю, и никакой руководитель не должен быть подчинен никакому руководителю.

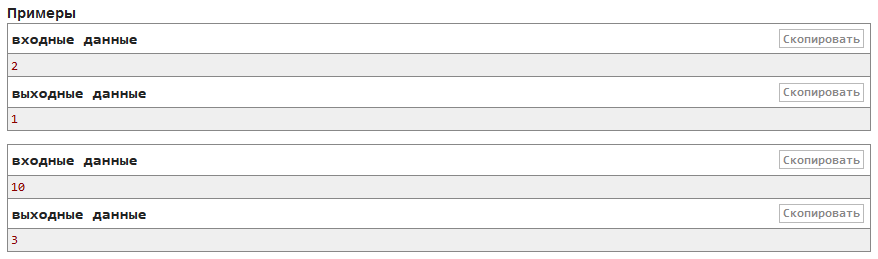
По данному числу сотрудников *n* найдите, сколькими способами Фафа может выбрать число руководителей *l*, чтобы можно было поровну разделить оставшихся сотрудников между ними.

**Входные данные**

В единственной строке содержится одно целое число *n* (2 ≤ *n* ≤ 105) — количество сотрудников в компании Фафы.

**Выходные данные**

Выведите одно целое число — ответ на задачу.



Во втором примере у Фафы есть три способа:

* выбрать только 1 руководителя и определить остальных 9 сотрудников под его руководство.
* выбрать 2-х руководителей, каждый будет руководить 4 сотрудниками.
* выбрать 5 руководителей, каждый будет руководить 1 сотрудником.

**Формирование команд**

В университете учатся *n* студентов. Количество студентов четно. Умение *i*-го студента программировать равно *ai*. Тренер хочет сформировать *n*/2 команд. Каждая команда должна состоять ровно из двух студентов, а каждый студент должен принадлежать ровно одной команде. Два студента могут сформировать команду только тогда, когда их умение программировать одинаково (иначе они не смогут понять друг друга и не смогут сформировать команду).

Студенты могут решать задачи, чтобы улучшать их умение программировать. Одна решенная задача увеличивает умение программировать на один.

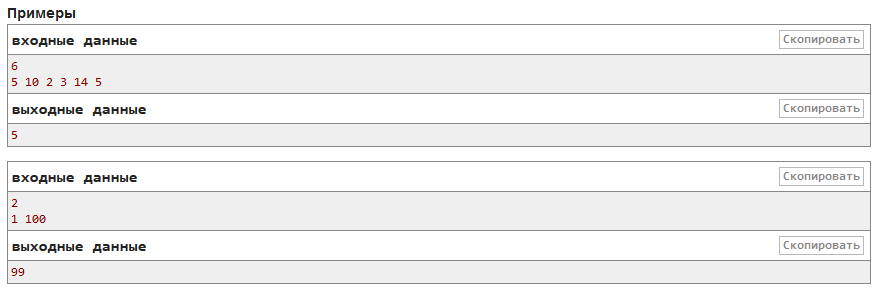
Тренер хочет знать, какое минимальное количество задач студентам необходимо решить, чтобы сформировать ровно *n/*2 команд (то есть каждая пара студентов должна формировать команду). Ваша задача — найти это количество.

**Входные данные**

Первая строка входных данных содержит одно целое число *n*(2≤*n*≤100) — количество студентов. Гарантируется, что *n*всегда является четным числом. Вторая строка входных данных содержит *n*целых чисел *a*1,*a*2,…,*an* (1≤*ai*≤100), где *ai* равно умению *i*-го студента программировать.

**Выходные данные**

Выведите одно целое число — минимальное количество задач, которое необходимо решить студентам, чтобы сформировать ровно *n/*2 команд.



*В первом тестовом примере* оптимальными будут команды: (3,4), (1,6) и (2,5), где числа в скобках являются номерами студентов. Тогда для того, чтобы сформировать первую команду, третий студент должен решить 1 задачу, чтобы сформировать вторую команду, никто не должен решать задачи, и чтобы сформировать третью команду, второй студент должен решить 4 задачи, таким образом ответ равен 1+4=5.

*Во втором тестовом примере* первый студент должен решить 99 задач, чтобы сформировать команду со вторым.

**Перчатка бесконечности**

Вы мельком увидели Таноса с Перчаткой бесконечности на руке. В перчатке есть места для шести камней бесконечности:

* камня силы (Power) пурпурного (purple) цвета,
* камня времени (Time) зеленого (green) цвета,
* камня пространства (Space) синего (blue) цвета,
* камня души (Soul) оранжевого (orange) цвета,
* камня реальности (Reality) красного (red) цвета,
* камня разума (Mind) желтого (yellow) цвета.

По цветам камней, присутствующим в перчатке, определите названия недостающих камней.

**Входные данные**

Первая строка содержит одно целое число *n* (0≤*n*≤6) — число камней в перчатке у Таноса. Следующие *n* строк содержат цвета камней в перчатке по одному в строке. Цвета обозначаются английскими словами из строчных букв: purple — фиолетовый, green — зеленый, blue — синий, orange — оранжевый, red — красный, yellow — желтый. Гарантируется, что все цвета различны.

**Выходные данные**

Выведите в первой строке одно целое число *m*(0≤*m*≤6) — число недостающих камней. Далее выведите *m* строк с названиями недостающих камней по одному в строке. Названия следует выводить английскими словами, первая буква — заглавная, остальные строчные: камень силы — Power, камень времени — Time, камень пространства — Space, камень души — Soul, камень реальности — Reality, камень разума — Mind. Названия могут быть выведены в любом порядке.



*В первом примере* у Таноса в перчатке уже есть камни Reality, Power, Mind и Soul, поэтому ему недостает двух камней Time и Space.

*Во втором примере* у Таноса в перчатке нет ни одного камня, поэтому ему недостает камней всех шести имеющихся видов.

**Виталий и ночь**

Как-то раз Виталий возвращался домой поздно ночью и заинтересовался вопросом: а сколько ещё людей не спит в это время? Для этого Виталий решил посмотреть, какие окна горят в доме, мимо которого он в этот момент проходил.

Виталий видит перед собой *n*-этажный дом и 2·*m* окон на каждом этаже. Согласно планировке, на каждом этаже располагается *m* квартир, пронумерованных от 1 до *m*, и каждой из них принадлежит два последовательных окна на данном этаже. Если пронумеровать окна от 1 до 2·*m* слева направо, то *j*-й квартире *i*-го этажа принадлежат окна 2·*j* - 1 и 2·*j* в соответствующем ряду окон. Виталий считает, что люди в квартире не спят, если хотя бы в одном из окон этой квартиры горит свет.

Используя информацию о свете в окнах данного дома, вы должны посчитать количество квартир, в которых, по мнению Виталия, жители не спят.

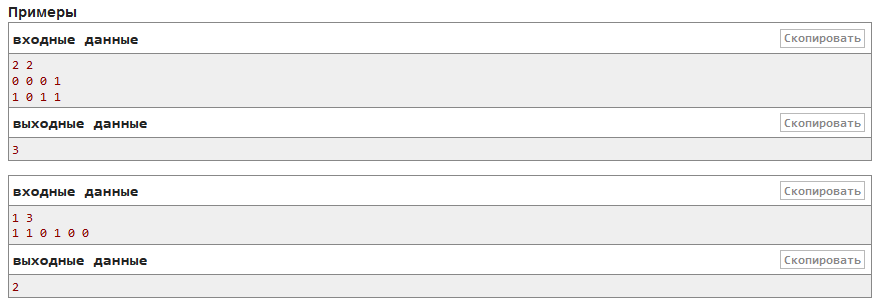
**Входные данные**

В первой строке входных данных следует два целых числа *n* и *m* (1 ≤ *n*, *m* ≤ 100) — количество этажей в доме и количество квартир на каждом этаже соответственно.

Следующие *n* строк описывают этажы сверху вниз и содержат по 2·*m* символов каждая. Если в *i*-м окне данного этажа горит свет, то в *i*-й позиции находится символ '1', иначе там находится символ '0'.

**Выходные данные**

Выведите единственно целое число — количество квартир, в которых горит свет хотя бы в одном окне, то есть, по мнению Виталия, там кто-то не спит.



*В первом тестовом примере* дом двухэтажный, по две квартиры на каждом этаже, то есть всего в нем 4 квартиры. Свет не горит только на втором этаже в левой квартире, так как в обеих комнатах этой квартиры свет выключен.

*Во втором тестовом примере* дом одноэтажный, и на первом этаже есть три квартиры. Свет горит в самой левой квартире (в обоих окнах) и в средней квартире (в одном окне). В правой же квартире свет не горит.