

МУНИЦИПАЛЬНЫЙ ЭТАП ВСЕРОССИЙСКОЙ ЛИМПИАДЫ ШКОЛЬНИКОВ ПО ИНФОРМАТИКЕ В 2020-2021 УЧЕБНОМ ГОДУ ДЛЯ УЧАЩИХСЯ 9-11 КЛАССОВ

Время проведения - 235 минут (3 часа 55 минут)

Максимальное количество баллов за задачу – 100

Максимальное количество баллов – 400

Особенности проведения - задания практического тура выполняются на компьютерах и сохраняются в отведенные папки. Не допускается использование локальной сети. Для проверки программ используется автоматическая тестирующая система Яндекс.Контест. Интернет-фильтр должен быть настроен только на использование соответствующих адресов и портов. Для авторизации должны быть подготовлены пароли и логины. Проверяющая система: <https://official.contest.yandex.ru/contest/XXXX/enter> (здесь XXXX - номер контеста).

Ограничение по времени – 1 секунда

Ограничение по памяти – 64Mb

Ввод данных – через стандартный поток ввода или из файла input.txt

Вывод данных – через стандартный поток вывода или в файл output.txt

Задача А. Планеты для заселения

Не так просто выбрать планету для колонизации. Есть множество параметров, которые влияют на пригодность планеты для жизни: температура поверхности, состав атмосферы, даже радиус и период вращения. Каждый параметр представляет собой в простейшем случае одно целое число. Планета пригодна для жизни, только если значения параметров не ниже их пороговых значений.

Комиссии по освоению космоса необходимо по ряду параметров выбрать подходящие для заселения планеты из списка. Далее среди выбранных планет нужно определить наиболее подходящие - те, суммарное значение параметров которых превосходит среднее арифметическое суммарных значений параметров для всех выбранных планет.

Входные данные

На первой строке через пробел записаны два целых числа: число планет $1 \leq n \leq 100$, число параметров $1 \leq k \leq 100$. На второй строке строке через пробел записано k целых чисел в диапазоне от 1 до 10^9 - пороговые значения параметров. Каждая из следующих n строк содержит следующую информацию: название планеты - строка, состоящая из не менее 1 и не более 100 заглавных букв латинского алфавита (названия не повторяются), далее через пробел k целых чисел, записанных через пробел - значения параметров для данной планеты.

Выходные данные

Необходимо вывести названия подходящих планет по одному на строке в порядке убывания суммарного значения параметров. Если суммарное значение параметров совпадает, названия планет располагаются в лексикографическом порядке (как в словаре).

Примеры

Входные данные	Выходные данные
4 3 1 2 2 MERCURY 1 2 1 VENUS 2 2 2 MARS 3 3 4 TAUKITAE 3 3 3	MARS TAUKITAE

Пояснения к примеру

Первая планета не подходит из-за недостаточных значений параметров. Для каждой из остальных суммарное значение параметров будет составлять 6, 10 и 9 соответственно. Среднее арифметическое равно 8,(3). Поэтому 3 и 4 планеты входят в итоговый список.

Задача В. Легенды космических битв

Последнее космическое сражение в обитаемой части Вселенной произошло десятки лет назад между армиями систем Тау Кита и Эпсилон Эридана.

Коварный предводитель Китян настроил искусственный интеллект Чрезвычайно Большой Пушки (на языке аборигенов *Overwhelmingly Massive Gun*, или сокращенно *OMG*) так, чтобы она нанесла максимальный урон флоту Эпсилон. Китянам удалось раскрыть тайный принцип расположения космических кораблей флота Эпсилон: самые ценные боевые единицы размещаются на расстоянии k друг от друга, считая от начала и до самого конца построения, на местах 1, $k+1$, $2k+1$ и т.д. При этом все корабли выстраиваются в прямую линию.

Например, если у Эпсилон есть корабли ценности 3, 2, 7, 10, 4, 5, а $k=2$, то их расставят, например, так: 7, 3, 10, 2, 5, 4. Наиболее ценные корабли (это оказались 7, 10 и 5) располагаются на расстоянии 2 друг от друга, начиная с самой первой позиции и до конца построения. Именно по этим позициям и будет наносить удары *OMG* Китян.

Но Совету старейшин Эпсилон Эридана стали известны планы Китян, и было принято решение изменить порядок размещения кораблей. В тот роковой момент исход сражения решил обычный программист, который помог Совету Эпсилон расположить корабли так, чтобы минимизировать полученный урон. Урон считается как сумма ценностей кораблей, по которым нанесет удар *OMG* Тау Кита.

Входные данные

На первой строке входного файла находятся через пробел два целых числа $1 \leq n, k \leq 10^4$. На следующей строке через пробел располагается n различных целых чисел в диапазоне от 1 до 10^9 - ценности кораблей.

Выходные данные

Необходимо вывести два различных варианта расположения кораблей - вариант ожидаемого китянами порядка (означающего максимальный урон) и вариант нового порядка, гарантирующего минимальный урон. Таким образом, в первой строке выходного файла необходимо перечислить через пробел ценности кораблей, соответствующие первому порядку. Во второй - второму. Если таких порядков может быть несколько - вывести любой из них.

Примеры

Входные данные	Выходные данные
6 2 3 2 7 10 4 5	7 3 10 2 5 4 3 7 2 10 4 5
7 3 1 2 3 4 5 6 7	5 4 3 6 2 1 7 1 5 6 2 7 4 3
2 100 1 2	2 1 1 2

Пояснения к примерам

В первом примере под ударом окажутся позиции 1, 3, 5, на которых должны располагаться наиболее ценные корабли 7, 10 и 5, что дает урон, равный 22. Но путем оптимального расположения кораблей можно добиться урона 9.

Во втором примере под ударом позиции 1, 4, 7, на которых должны располагаться корабли 5, 6, 7. Новое расположение сводит урон к 6.

В третьем примере под ударом только одна первая позиция.

Задача С. Космические Корабли Цивилизации

Когда-то Древняя и таинственная цивилизация принесла в дар людям первый космический корабль - необычное живое существо, именуемое Парамартха, на чьей огромной спине люди отправлялись в первые дальние космические полеты. Но запаса энергии Парамартхи не хватало на межзвездные путешествия.

И спустя время Парамартха произвела на свет первую Брингу, которая могла запasti в 2 раза больше энергии, чем Парамартха. Вторая Бринга могла запастись в 3 раза больше энергии, чем сама Парамартха. За долгие годы человечество смогло узнать эту невероятную математику: каждая следующая Бринга, рождавшаяся у Парамартхи, могла запasti

количество энергии, большее Парамартхи в число раз, равное очередному по счету простому числу.

Оказалось, что и каждая Бринга начинала порождать новые Бринги. И они тоже могли запасать в 2, 3, 5, 7, 11 и так далее раз больше энергии, чем материнская Бринга. Этот процесс продолжался тысячи лет.

Стало понятно, что начиная с какого-то количества энергии, запасаемой космическим кораблем, становится возможен межзвездный перелет, выполняемый на сверхсветовых скоростях. Однако слишком большой запас энергии делал корабль тяжелым и неповоротливым, что не позволяло выполнять перелеты безопасно. Стало важно научиться определять, сколько космических кораблей может запасти заданное число единиц энергии. Для простоты можно считать, что Парамартха была способна запасти 1 единицу энергии.

Часть генеалогического дерева космических кораблей выглядит как показано на рисунке.

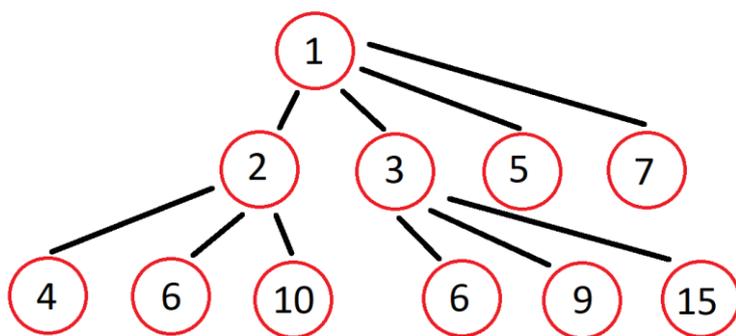


Рис. 1

Входные данные

Единственная строка входного файла содержит одно целое число в диапазоне от 1 до 1000 - количество единиц энергии N.

Выходные данные

Необходимо вывести одно целое число - количество кораблей, обладающих N единиц энергии.

Примеры

Входные данные	Выходные данные
1	1
4	1
6	2

Задача D. Галактическая работа в команде

В галактическом Совете настало время укрепить командный дух. Каждый участник галактического Совета дружит с какими-то участниками (отношения дружбы всегда взаимны), а с другими - в достаточно нейтральных отношениях. Но Галактика на пороге больших вызовов, и дружеские доверительные отношения в Совете - необходимое условие выживания цивилизации.

Для максимальной пользы дела, необходимо разбить огромный Совет на две большие не обязательно равные по числу участников команды такие, что в каждой команде отсутствуют отношения дружбы между любыми двумя ее участниками. Можно ли это сделать?

Входные данные

Первая строка входного файла содержит через пробел два целых числа: количество участников Совета $2 \leq n \leq 100$, количество дружеских связей $1 \leq m \leq 100$. На каждой из следующих m строк записана через пробел пара целых чисел: номера двух участников совета, связанных отношениями дружбы. Участники нумеруются целыми последовательными числами, начиная с 1.

Выходные данные

Вывести “yes” (без кавычек), если можно разбить Совет требуемым образом, или “no” (без кавычек), если разбить Совет невозможно.

Примеры

Входные данные	Выходные данные
3 2 1 2 2 3	yes
3 3 1 2 2 3 3 1	no