

Контрольная точка 3

Необходимо составить отчет, содержащий для каждой задачи следующие пункты:

1. Название и формулировка задачи.
2. Листинг программы (на одном из языков программирования по выбору: C++, C#, Java или Python) с комментариями, поясняющими ее работу. Программы для заданий №1-4 необходимо составить таким образом, чтобы входные данные считывались из текстового файла, формат которого указан в заданиях.
3. Для задний №1-4 привести по пять тестов, демонстрирующих корректность работы программы при различных исходных данных. Тесты оформляются в виде таблицы:

Стандартный ввод	Стандартный вывод	Время выполнения (мс)
------------------	-------------------	-----------------------

Для задания №5 результат указать в виде:

- Количество ходов наиболее продолжительной партии и ее результат (кто победил) при сильнейшей игре обоих игроков.
 - Все значения (S_1, S_2) первоначального количества камней в обеих кучах, при которых партия продолжается максимальное количество ходов при сильнейшей игре обоих игроков.
 - Время выполнения программы (в мс)
4. Расчет с пояснениями вычислительной сложности алгоритма, по которому составлена программа.

Важно!!!: оценивается только отчет; скриншоты программ и результатов их работы не рассматриваются и не оцениваются.

Задание № 1. Поиск множеств

Ограничение по времени: 1 секунда. Ограничение по памяти: 64 мегабайта

В первой строке файла содержится три числа: N – количество эталонных множеств, M – размер каждого из множеств и K – количество пробных множеств.

Каждое из множеств содержит целые числа от 0 до 10^9 , числа могут повторяться.

Требуется для каждого из пробных множеств вывести в отдельной строке цифру '1', если это множество в точности совпадает с каким-либо из эталонных множеств и цифру '0', если оно ни с одним не совпадает, то есть выведено должно быть в точности K строк.

$$5 \leq N \leq 50000$$

$$3 \leq M \leq 1000$$

$$5 \leq K \leq 50000$$

Примеры

Файл входных данных	Стандартный вывод
10 3 5	1
6 5 1	1
7 9 3	0
2 3 2	0
7 2 9	1
9 6 2	
6 6 6	
9 4 1	
8 4 4	
8 3 2	
1 2 6	
9 7 2	

1 6 5 3 7 7 4 4 6 3 9 7	
10 7 5 8 4 0 3 6 9 2 3 5 0 4 3 1 1 7 1 0 3 1 2 4 7 1 5 1 5 5 1 3 4 0 0 3 4 0 3 3 3 6 3 9 3 3 4 1 3 1 8 1 1 1 6 8 6 8 2 5 6 8 1 3 9 3 7 5 7 1 4 0 3 1 1 3 3 8 4 1 2 1 1 6 6 8 8 1 1 5 7 5 1 5 3 4 1 3 1 1 8 0 0 1 2 8 2 6	1 1 1 1 0

Задание №2. Анаграммы

Ограничение по времени: 0.5 секунд. Ограничение по памяти: 256 мегабайт

Как известно, анаграммами называются слова, которые могут получиться друг из друга путем перестановки букв, например LOOP, POOL, POLO. Будем называть все слова такого рода *комплексом*.

На вход программы подается число слов $1 \leq N \leq 100000$. В каждой из очередных N строк присутствует одно слово, состоящее из заглавных букв латинского алфавита. Все слова имеют одинаковую длину L: $3 \leq L \leq 10000$.

Требуется определить число комплексов во входном множестве.

Формат входных данных:

N
W1
W2
...
WN

Формат выходных данных:

Количество_комплексов

Пример

Файл входных данных	Стандартный вывод
5 LOOP PLLO POLO POOL OLPL	2

Задание №3. Точные квадраты

Ограничение по времени: 1 секунда. Ограничение по памяти: 256 мегабайта

Можете ли вы по десятичному представлению натурального числа определить, является ли это число точным квадратом? А если в числе много десятичных знаков (до 300 цифр)?

Формат входных данных:

Первая строка содержит N — количество чисел, которые нужно проверить. $5 \leq N \leq 10^6$. В последующих N строках — десятичные представления натуральных чисел количеством десятичных цифр в представлении не более 300.

Формат выходных данных:

Для каждого из чисел, являющихся точным квадратом, вывести его номер. Нумерация начинается с единицы.

Пример

Файл входных данных	Стандартный вывод
5	2
63	3
31634752772484	5
244364259903929646968330496	
244364259903929646968330396	
16	

Задание №4. Магараджа

Ограничение по времени: 60 секунд. Ограничение по памяти: 100 мегабайт

Магараджа — это шахматная фигура, сочетающая возможности ферзя и коня. Таким образом, магараджа может ходить и бить на любое количество клеток по диагонали, горизонтали и вертикали (т.е. как ферзь), а также либо на две клетки по горизонтали и на одну по вертикали, либо на одну по горизонтали и на две по вертикали (как конь). Необходимо найти максимальное число фигур магараджей, которое можно расставить на квадратной доске размером $N \times N$ так, чтобы они не били друг друга, и отобразить одну из таких расстановок.

Входные данные: натуральное число N ($N < 30$).

Выходные данные: в первой строке — максимально возможное число фигур, в последующих строках — вариант расстановки магараджей.

Пример

Стандартный ввод	Стандартный вывод (М – клетка, занятая фигурой магараджи, точка – пустая клетка):
5	4 .M...M M.... ...M.

Задание №5. Игра «Две кучи»

Ограничение по времени: 10 секунд. Ограничение по памяти: 20 мегабайт

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) один камень или увеличить количество камней в куче в два раза.

Например, пусть в одной куче 10 камней, а в другой 5 камней; такую позицию в игре будем обозначать $(10, 5)$. Тогда за один ход можно получить любую из четырёх позиций: $(11, 5)$, $(20, 5)$, $(10, 6)$, $(10, 10)$. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 77. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, при которой в кучах будет 77 или больше камней.

В начальный момент в первой куче было S_1 камней, во второй куче — S_2 камней; $1 \leq S_1, S_2 \leq 69$.

Каждый игрок играет сильнейшим образом, т.е. если он может выиграть, то старается это сделать за наименьшее число ходов. А если игрок не может выиграть — то он старается максимально увеличить количество ходов в партии.

Определить, при каком начальном количестве камней в кучах $(S_1; S_2)$ партия будет продолжаться в этих условиях максимальное количество ходов. Если это максимальное количество ходов достигается при различных начальных значениях камней в кучах, то указать все варианты.