



Z01. Выравнивание команд.

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Для проведения различных игр и конкурсов на военно-патриотическом слете среди лучших школьников организуются команды с равным количеством участников. После отъезда части школьников количество участников в командах стало различным. После очередного заезда школьников было решено выровнять количество участников в командах. Определите, возможно ли распределить вновь приехавших ребят среди имеющихся команд так, чтобы количество участников в командах было равным. Причем каждую команду должен пополнить хотя бы один из вновь из прибывших школьников.

В первой строке число задается N ($2 < N < 100$) – количество команд. В следующей строке через пробел задаются количества человек в каждой команде (не более 300). В последней строке задается количество вновь прибывших школьников (не более 10000).

Если можно создать команды с равным количеством участников, то необходимо вывести количество участников в одной команде. Если такое распределение не возможно, то вывести «NO».

Входные данные	Результат
5 11 12 13 14 15 20	17
3 10 15 20 15	NO

Z02. Частичные простые.

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Подстрока числа формируется стоящими подряд цифрами исходного числа. Например, число 6158 содержит подстроки 6, 1, 5, 8, 61, 15, 58, 615, 158, 6158. Вы должны найти самую большую подстроку числа, которая будет являться простым числом.

Входные данные: Входные данные содержат число N ($0 \leq N \leq 1,000,000,000$).

Выходные данные: В выходные данные вывести самую большую подстроку N , которая будет являться простым числом или вывести "0" (без кавычек), если нет ни одной такой подстроки.

Пример входных и выходных данных

Входные данные:	Выходные данные
2319	31
6804	0

Z03. Сортировка времени

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

В входных данных записано сначала число N ($1 \leq N \leq 100$), а затем N моментов времени. Каждый момент времени задается 3 целыми числами - часы (от 0 до 23), минуты (от 0 до 60) и секунды (от 0 до 60).

В выходные данные выведите моменты времени, упорядоченные в порядке возрастания (момент времени также выводится в виде трех чисел, ведущие нули выводить не обязательно)

Входные данные	Выходные данные
4	7 30 00
10 20 30	10 20 30
7 30 00	13 30 30
23 59 59	23 59 59
13 30 30	

Z04. Площадь многоугольника.

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

В входных данных записано число N ($1 \leq N \leq 100$), а затем N пар чисел, координаты точек многоугольника по обходу против часовой стрелки. Координаты точек в диапазоне от -1000 до 1000.

В выходные данные вывести площадь данного многоугольника с точностью до 3 цифр в дробной части.

Входные данные	Выходные данные
4	16.000
0 0	
4 0	
4 4	
0 4	

Z05. Простые делители

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Найти все простые делители натурального числа N .

Входные данные: Ввести одно число N ($1 \leq N \leq 200000000$).

Выходные данные: Вывести все простые делители натурального числа N в порядке их возрастания.

Пример входных данных:	Пример выходных данных
10	2 5
1000	2 5

Z06. Словарь

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

У Васи на клавиатуре не работает клавиша пробел. Поэтому все тексты он теперь набирает слитно. Напишите программу, которая будет разделять набранный Васей текст на слова из данного словаря.

Входные данные: Во входных данных в первой строке записан текст, введенный Васей – одна строка из не более чем 100 латинских строчных букв. В следующей строке записано количество слов в словаре N – натуральное число, не превосходящее 2000. В следующих N строках записаны

слова из словаря – по одному слову в каждой строке, каждое слово длиной не более 20 латинских строчных букв. Слова записаны в алфавитном порядке.

Выходные данные: выведите Васин текст с пробелами между словами (пробел после последнего слова допустим). Если возможно несколько вариантов разбиения строки на слова, выведите один любой их них. Гарантируется, что хотя бы один способ разбиения строки на словарные слова существует.

Входные данные	Выходные данные
Whatcanido б а an can do i	What what can I do
Vababababa 2 а b	b a b a b a b a

Z07. "Счастливым билет"

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Билет называется счастливым, если его можно разрезать прямой линией на две части таким образом, что оказавшиеся на них числа имеют одинаковые цифровые корни. Чтобы вычислить цифровой корень числа, его цифры складывают, если в результате получится число большее или равное 10, то цифры складывают снова и так далее, пока не получится число от 0 до 9 – это и есть цифровой корень. Например, билет с номером 0015420 является счастливым, так как разрезав его на части с числами 0015 и 420 имеем у этих чисел одинаковые цифровые корни. Написать программу, которая определит является ли счастливым билет с заданным номером.

Входные данные: Входные данные содержит номер счастливого билета. Номер может начинаться с нулей и содержит не более 100 цифр.

Выходные данные: Выходной данные должен содержать 1, если билет счастливый и 0 иначе.

Входные данные	Выходные данные
0015420	1
00100	0

Z08. "Последовательность ИЮ"

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Последовательность 011212201220200112... строится следующим образом: сначала пишется 0, затем повторяется следующее действие: уже написанную часть приписывают справа с заменой 0 на 1, 1 на 2, 2 на 0, и т.д.

Написать программу, которая по заданному натуральному числу n определяет, какое число стоит на n-ом месте.

Входные данные: Входные данные содержит число n ($1 \leq n \leq 2147483647$).

Выходные данные: Выходные данные должен содержать одно искомое число.

Входные данные	Выходные данные
10	2

Z09. "Нить Краснова"

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Тезею из лабиринта Минотавра помог выйти клубок ниток. Вы можете вместо клубка использовать персональный компьютер.

Написать программу, которая вводит маршрут Тезея в лабиринте и находит обратный путь, по которому Тезей сможет выйти из лабиринта, не заходя в тупики и не делая петель.

Входные данные: Входные данные содержит маршрут Тезея, который представлен строкой, состоящей из букв: N, S, W, E и длиной не более 200.

Буквы означают:

N - один "шаг" на север,

S - один "шаг" на юг,

W - один "шаг" на запад,

E - один "шаг" на восток.

Выходные данные: В выходные данные записывается аналогично входным данным обратный путь.

Входные данные	Выходные данные
EENNESWSSWE	NWW

Z10. "Оставшееся число Маши"

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Задан ряд последовательных натуральных чисел от n до m ($n < m$), из которого удаляют сначала все числа, стоящие на нечетных местах. Затем из оставшегося ряда удаляют все числа, стоящие в нем на четных местах. Эти действия повторяют до тех пор, пока не останется одно число.

Написать программу, которая находит оставшееся число.

Входные данные: Во входных данных, в первой строке записано первое число n , а во второй – второе m ($n < m < 1000000000$).

Выходные данные: В выходные данные вывести оставшееся число.

Входные данные	Выходные данные
1	2
4	

Z11. Значение выражения.

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Дана строка, состоящая не более чем из 25000 символов, изображающая арифметическое выражение вида «<цифра> \pm <цифра> \pm ... \pm <цифра>», где на месте знака операции « \pm » находится символ «+» или «-» (например, «4+7-2-8»). Вывести значение данного выражения (целое число).

Входные данные: Входные данные содержит строку, изображающую арифметическое выражение.

Выходные данные: В выходные данные вывести одно число, значение данного выражения.

Входные данные	Выходные данные
1+1+1+1+1-5	0
1-2+3-4+5-6+7-8+9	5

Z12. Составление наибольшего числа

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Дано натуральное десятичное число, содержащее до 50 разрядов. Составить из этого числа наибольшее число, которое делится на 4. Если такое число составить невозможно, то выведите NO.

Входные данные	Результат
3033176577	7775333016
13789137	NO

Z13. Количество слов

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Во входном файле записана строка текста, в которой могут встречаться:

- прописные и строчные (т.е. большие и маленькие) латинские буквы,
- пробелы,
- знаки препинания: точка, запятая, восклицательный и вопросительный знак,
- символ `-`, обозначающий в некоторых случаях тире, а в некоторых — дефис.

Слово — это последовательность подряд идущих латинских букв и знаков дефис, ограниченная с обоих концов. В качестве ограничителей могут выступать начало строки, конец строки, пробел, знак препинания, тире. Тире отличается от дефиса тем, что слева и справа от знака дефиса пишутся буквы, а хотя бы с одной стороны от тире идет либо начало строки, либо конец строки, либо пробел, либо какой-либо знак препинания, либо еще одно тире.

Напишите программу, определяющую, сколько слов в данной строке текста.

Входные данные: Во входных данных записана строка длиной не более 200 символов.

Выходные данные: В выходные данные выведите одно число — количество слов, которые содержатся в исходной строке.

Входные данные	Выходные данные
Hello , world!	2
www.olympiads.ru	3
Gyro-compass – this is a ...	4

Z14. Цифры и буквы

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Напишите программу, определяющую, требуется ли для записи заданного числа в шестнадцатеричной форме использовать только буквы, только цифры, или буквы и цифры одновременно.

Входные данные: В входных данных содержится несколько строк. В каждой строке содержится одно целое число N ($1 \leq N < 10^9$) в десятичной форме. Строка, содержащая число 0, служит признаком конца ввода и не обрабатывается.

Выходные данные: В выходные данные для каждого числа из входного файла вывести на отдельной строке цифры 1 - «Только буквы», 2 - «Только цифры», 3- «Буквы и цифры», в зависимости от множества символов, использованных для шестнадцатеричного представления числа.

Входные данные	Выходные данные
----------------	-----------------

256	2
10	1
28	3
0	

Z15. Экономия топлива

Входные данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Вы собираетесь поехать на машине за город на пикник, но у вас ограниченный запас топлива. Вы знаете, сколько топлива в час расходует автомобиль, если ехать на определенной скорости. Напишите программу, которая определяет максимальное расстояние, на которое можно уехать с заданным количеством топлива. Поездка должна происходить при постоянной скорости, одной из тех, которая задана.

Входные данные: В первой строке входных данных содержится целое число – количество наборов исходных данных K ($1 \leq K \leq 10$). Далее следует K блоков, каждый блок описывает один набор. В первой строке блока содержатся два целых числа, разделенных пробелом – количество возможных скоростных режимов вашего автомобиля N ($0 < N \leq 50$) и запас топлива F в миллилитрах ($100 \leq F \leq 50000$). Далее следует N строк, в каждой строке содержатся два целых числа, разделенных пробелом – скорость вашего автомобиля S_i в километрах в час ($5 \leq S_i \leq 250$) и расход топлива в миллилитрах за час C_i ($1000 \leq C_i \leq 20000$) при скорости S_i .

Выходные данные: В выходные данные для каждого набора вывести строку, содержащую одно вещественное число с тремя десятичными знаками – максимальное расстояние, которое автомобиль сможет проехать с заданным количеством топлива.

Входные данные	Выходные данные
2	100.000
1 10000	1100.000
100 10000	
3 40000	
100 4000	
60 5000	
110 4000	

Z16. Простые числа

Входные данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Дан набор различных натуральных чисел. Будем называть число “простым для заданного набора”, если число не делится ни на одно из чисел набора, кроме самого себя.

Входные данные: Во входных данных в первой строке содержится целое число N ($1 \leq N \leq 100$) – количество чисел в наборе. Во второй строке содержатся N различных целых чисел от 1 до 1000000, разделенных пробелами.

Выходные данные: В выходные данные вывести “простые для заданного набора” числа, разделяя числа одним пробелом. Числа выводятся в том порядке, в котором они шли во входном файле.

Входные данные	Выходные данные
6	5 3 8
10 5 3 15 6 8	

Z17. Забор

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Однажды тете Полли в голову пришла мысль сделать дома ремонт. Начать она решила, конечно, с забора. "В прошлый раз ты прекрасно справился с работой", – сказала тетья Полли Тому Союеру, – "но сейчас модно красить заборы не одним цветом, а несколькими цветами, чтобы получился красивый узор. Вот тебе банки с краской, вот кисточка, вот образец из журнала, как надо покрасить забор".

Том Соьер вышел на улицу и с тоской оглядел огромный забор. Из-за угла появился Билли Гейл с удочкой и ведерком. "Что, старик, работать приходится?" – спросил Билли. Том обернулся. "А ты хотел бы немножко покрасить?" "А что ты мне за это дашь?" Том понял, что Билли – единственный мальчик в городе, который не будет платить за удовольствие покрасить забор. К счастью, у Тома оставалось еще много вещей, полученных за предыдущую покраску забора. "У меня есть бумажный змей, дохлая крыса на веревочке, губная гармоника, ржавый ключ, ..." "Хорошо", – прервал его Билли, – "я согласен, но ты будешь отдавать мне одну вещь за каждую непрерывную полосу забора, покрашенную одним цветом". Том решил придумать такой способ покраски забора, чтобы отдать как можно меньше вещей алчному Билли. Например, образец ААВВСВВАА можно покрасить тремя полосами. Первый раз макаем кисть в краску А и красим весь забор, получается АААААААААА. Далее макаем кисть в краску В и красим середину забора, получается ААВВВВВВАА. И последний раз макаем в краску С и красим одну доску в самой середине, получается ААВВСВВАА.

Входные данные: В первой строке входных данных содержится число N ($1 \leq N \leq 20$) – количество образцов раскраски. В следующих N строках даны образцы раскраски забора. Каждая строка содержит от 1 до 200 прописных латинских букв. Одна буква соответствует одной доске забора. Различные цвета обозначены различными буквами.

Входные данные: В выходные данные для каждого образца раскраски, заданного тетей Полли, вывести на соответствующей отдельной строке минимальное количество вещей, которое Том Соьер должен отдать Билли, чтобы раскрасить забор по этому образцу.

Входные данные	Входные данные
2	3
ААВВСВВАА	4
АВСD	

Z18. Игра

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Двое играют в следующую игру. На столе лежит кучка камешков. Игроки ходят по очереди, беря из кучки 1 или более камней, но не более половины числа оставшихся камней. Если остался только один камень, то игрок, делающий очередной ход, проигрывает. Напишите программу, определяющую, можно ли выиграть в текущей игровой ситуации, и если да, то какое количество камней нужно взять очередным ходом, чтобы выиграть при безошибочной игре соперников.

Входные данные: Во входных данных содержится одно целое число N ($1 \leq N \leq 10^9$) – количество камней в кучке.

Входные данные: В выходные данные вывести одно целое число – количество камней, которое нужно взять из кучки, чтобы выиграть при безошибочной игре соперников. Если выигрыш в этой ситуации невозможен, то вывести число 0.

Входные данные	Выходные данные
4	1
3	0

Z19. Исчезнувшие числа Фибоначчи

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Ряд чисел Фибоначчи, был перемешан и передан по компьютерной сети. Оказалось, что при передаче этого ряда пропало два числа из ряда Фибоначчи, неравных 1. Найдите эти два числа и выведите часть ряда чисел Фибоначчи от большего к меньшему числу.

Максимальное количество переданных чисел – 40. Использовать не более 8 переменных длинного целого типа.

Входные данные	Результат
2 1 3 1 8 21	13 8 5
1 1 3 8 13	21 13 8 5

Z20. Магический квадрат

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

0	13	7	20
-9	4	-2	11
12	25	19	32
3	16	10	23

Квадрат на рисунке обладает интересным магическим свойством. Выберите любое из чисел, вписанных в клетки квадрата, и обведите кружком. Вычерните все числа, стоящие в той же самой строке и столбце, что и выбранное число. Снова выберите число из еще не вычеркнутых или не обведенных чисел и обведите его кружком, и снова вычерните числа, стоящие в той же строке и столбце, что и выбранное во второй раз число. Прodelайте эту процедуру в третий раз, получив третье число.

Обведите кружком оставшееся четвертое число. Сложите все четыре числа, обведенных кружком. Их сумма будет равна 46. Сколько бы вы раз не повторяли эксперимент, сумма всегда будет одной и той же, равной 46 – сумме чисел на диагонали квадрата.

Напишите программу, которая по числам, расположенным на главной диагонали квадрата, строит квадрат, обладающий указанным магическим свойством.

Входные данные: Во входных данных в первой строке содержится целое число N ($1 < N \leq 100$) – размер квадрата, далее следует N строк; i -ая строка ($1 \leq i \leq N$) содержит одно целое число a_{ii} ($|a_{ii}| < 1000$) – число, которое должно быть на главной диагонали магического квадрата. Все числа, расположенные на диагонали квадрата, различны.

Выходные данные: В выходные данные вывести N строк по N целых чисел, разделенных пробелом – магический квадрат. Числа в квадрате не должны повторяться и не должны по абсолютному значению превосходить 10^6 . Если возможно несколько вариантов заполнения, то нужно вывести любой (один) из них.

Входные данные	Выходные данные
4	0 13 7 20
0	-9 4 -2 11
4	12 25 19 32
19	3 16 10 23
23	

Z21. Вырежем шахматную доску.

Из прямоугольника размером M на N , клетки которого раскрашены в черный и белый цвет, требуется вырезать квадрат размера K на K , раскрашенный в шахматном порядке (то есть любые две соседние клетки раскрашены в разные цвета).

Входные данные: В первой строке входных данных записаны числа M, N, K ($1 \leq K \leq M, N \leq 900$) через пробел. В последующих M строках расположены N чисел 0 или 1, задающие цвета клеток прямоугольника.

Выходные данные: В выходные данные вывести в единственную строку номера строки и столбца клетки – левого верхнего угла вырезаемой доски. Числа должны разделяться пробелом. Если требуемый квадрат вырезать невозможно, вывести одно число 0.

Входные данные	Выходные данные
3 4 2 1 1 1 0 1 1 1 0 1 1 0 1	2 3

Z22. Морской бой

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Поле для игры в морской бой имеет размер $N \times N$ клеток. Клетки, занятые кораблями, содержат символ «X» (ASCII 88), а пустые клетки — символ «.» (ASCII 46). Корабли не соприкасаются друг с другом. Требуется определить количество кораблей длиной L клеток.

Входные данные: В первой строке входных данных содержатся числа N и L ($1 \leq N, L \leq 20$).

Следующие N строк по N символов каждая содержат описание поля.

Выходные данные: Выходной данные должен содержать единственное целое число — количество L -клеточных кораблей.

Входные данные	Выходные данные
3 4 2 1 1 1 0 1 1 1 0 1 1 0 1	2 3

Z23. Вынутый разворот

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Брошюра составлена из листов. На каждой стороне листа напечатано по две страницы. Страницы пронумерованы начиная с первой.

Из брошюры был вынут один лист. Требуется по двум номерам страниц, напечатанным на одной из сторон этого листа, определить общее количество страниц в брошюре.

Входные данные: Во входных данных содержатся два целых числа A и B ($1 \leq A, B \leq 10000$) — номера страниц на стороне листа, в произвольном порядке.

Выходные данные: Выходные данные должен содержать число страниц либо 0, если такой брошюры не существует.

Входные данные	Выходные данные
9 1	0
3 14	16

Z24. Мнутная стрелка

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Центр циферблата часов имеет координаты (0,0), а конец минутной стрелки - координаты (x, y). Ось ординат направлена вверх. Требуется определить, сколько полных минут показывают часы.

Входные данные: Входные данные содержит два вещественных числа, разделённых пробелом - x y .

Выходные данные: Выходные данные должен содержать единственное целое число в диапазоне от 0 до 59 -- число минут.

Ограничения: Координаты x, y не равны одновременно нулю.

Входные данные	Выходные данные
-587.3 -834	36

Z25. Склеенные числа

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Склеиванием чисел назовем выписывание этих чисел одного за другим без пробелов. Например, склеив числа 15 и 71, получим 1571. Если склеивать подряд все натуральные числа, получится бесконечная последовательность цифр 12345678910111213141516171819202122...

По данному числу N ($1 \leq N \leq 10000$) определить N -ую по счету цифру в этой последовательности.

Входные данные: Входные данные состоит из одного числа N .

Выходные данные: Выходные данные должен содержать одно число N -ую цифру в этой последовательности.

Входные данные	Выходные данные
1	1
112	0

Даны две текстовые строки. Требуется найти самую длинную подстроку первой строки, составленную из второй.

Входные данные: В первой строке входных данных содержится первая строка, во второй — вторая строка. Длина каждой строки — от 1 до 255 символов.

Выходные данные: Выходные данные должен содержать одну самую длинную подстроку. Если найдено несколько подстрок одинаковой длины, следует вывести ту, которая встречается раньше. Входные данные таковы, что первая строка обязательно содержит хотя бы один символ из второй.

<i>Входные данные</i>	<i>Выходные данные</i>
dabcxxdxb xbbc	bcx

Z29. Числа и музыка

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

“Музыкальная” последовательность, состоящая из нулей и единиц, строится следующим образом: на первом шаге последовательность состоит из одной единицы. На каждом следующем шаге к последовательности приписывается ее инверсия, т. е. последовательность, в которой 1 заменены на 0 и наоборот.

Таким образом, на первых 5 шагах последовательность будет выглядеть так:

1
10
1001
10010110
1001011001101001

Требуется по данному числу N ($1 \leq N \leq 1000$) определить значение N -го элемента последовательности.

Входные данные: Во входных данных содержится целое число N .

Выходные данные: В выходных данных должно содержаться число 0 (нуль) либо 1.

<i>Входные данные</i>	<i>Выходные данные</i>
13	1

Z30. Железные дороги

Входной данные: ввод с клавиатуры

Выходные данные: на консоль

Время тестирования: 1 сек.

Ограничение памяти: 64Мб.

Инфраструктура железнодорожной сети некоторой страны состоит из некоторого количества индивидуальных железнодорожных сегментов. Каждый сегмент соединяет два различных города. Нам известны длины всех сегментов, и сеть такова, что между любыми двумя городами существует единственный путь. Недавно, руководство железнодорожной компании решило заменить свои старые поезда ультрамодными и быстрыми поездами, импортированными из Японии. Маршрут поезда состоит из последовательности из двух или более городов, причем все соседние города в маршруте соединены сегментами. Длина маршрута определяется как сумма длин всех сегментов в нем. Так как поезда очень быстрые, то не разрешается, чтобы два различных маршрута проходили через один и тот же город.

Напишите программу, которая определит новые маршруты в сети, удовлетворяющие следующим условиям: ровно один маршрут проходит через каждый город и суммарная длина маршрутов максимальна.

Входные данные: Первая строка ввода содержит целое число N , $1 \leq N \leq 2000$, количество городов. Города нумеруются целыми числами от 1 до N . Следующие $N-1$ строк содержат описания сегментов. Каждая строка содержит три целых числа A , B и C , обозначающие сегмент между городами A и B , имеющий длину C , $1 \leq C \leq 1000000$.

Замечание.

Входные данные будут таковы, что решение всегда существует.

Выходные данные: Первая и единственная строка вывода должна содержать искомую максимальную сумму длин маршрутов.

Входные данные	Выходные данные
5 1 2 10 1 3 10 1 4 10 4 5 1	21
6 1 2 1 1 3 3 1 4 7 2 5 4 2 6 6	20
9 1 2 1 2 4 2 3 4 1 4 5 5 6 5 1 5 7 2 7 8 1 8 9 2	11