

Интеллектуальная олимпиада ПФО среди школьников
Задания муниципального этапа по программе «Программирование»
2021-2022 учебный год

Время выполнения – 5 часов

Максимальное количество баллов за решение одной задачи – 100

Задача А. Робот

Имя входного файла: `input.txt`

Имя выходного файла: `output.txt`

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Робот находится в точке плоскости с целыми координатами (X, Y) , где $(0 \leq X, Y \leq 20; X + Y > 0)$. Он должен достичь цели, которая находится в начале координат. Каждый ход робота – продвижение на единицу по горизонтали (налево или направо) или вертикали (вверх или вниз). Если робот достигает цели, то немедленно останавливается.

Необходимо найти количество способов достижения цели ровно за Z шагов $(1 \leq Z \leq 20)$.

Формат входных данных

В единственной строке находятся через пробел три числа X, Y и Z .

Формат выходных данных

В единственной строке вывести количество способов достижения цели ровно за Z шагов.

Примеры:

№	input.txt	output.txt	Комментарий
1	1 0 1	1	
2	0 1 3	5	Возможны следующие варианты движения робота: $(0, 1) - (0, 2) - (0, 1) - (0, 0)$ $(0, 1) - (1, 1) - (0, 1) - (0, 0)$ $(0, 1) - (1, 1) - (1, 0) - (0, 0)$ $(0, 1) - (-1, 1) - (0, 1) - (0, 0)$ $(0, 1) - (-1, 1) - (-1, 0) - (0, 0)$

Задача В. Марсианский прибор

Имя входного файла: input.txt

Имя выходного файла: output.txt

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Посещая выставку достижений марсианского народного хозяйства, Петя увидел странный аппарат. У аппарата было восемь кнопок. Некоторые были нажаты, а некоторые отжаты. Гид объяснил Пете, что это прибор для усиления интеллекта (IQ). Любой марсианин с помощью этого прибора может поднять уровень своего интеллекта до определенного значения. Петя поинтересовался, до какого именно значения прибор увеличит его IQ? Вот что ему рассказал гид:

- изначально прибор считает интеллект Пети равным единице;
- затем прибор анализирует состояние кнопок (слева направо), и если очередная кнопка нажата, то прибор удваивает значение IQ и добавляет единицу; если кнопка отжата, то прибор просто удваивает значение IQ;
- полученный результат становится уровнем интеллекта Пети.

Посмотрев на кнопки и сосчитав в уме уровень своего будущего IQ (он оказался равен N), Петя хотел было воспользоваться прибором, но гид его предостерег:

– Все, что было рассказано о расчете IQ справедливо только для марсиан! А для людей алгоритм отличается тем, что если кнопка нажата и ее порядковый номер (если считать слева) четный, то прибор удваивает значение IQ и добавляет единицу, а если номер нечетный, то прибор удваивает IQ и вычитает единицу. Для отжатых кнопок разницы между марсианами и людьми нет.

Помогите Пете определить, какой уровень IQ будет у него, если он все-таки воспользуется прибором.

Формат входных данных

В файле содержится целое число N .

Формат выходных данных

В выходной файл необходимо вывести значение IQ, которое получит Петя, если воспользуется прибором.

Примеры:

№	input.txt	output.txt
1	300	220
2	256	256
3	500	180

Задача С. Цифры в числах

Имя входного файла: input.txt

Имя выходного файла: output.txt

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Как-то раз студент Вася не мог заснуть. Сначала он представлял себе караван верблюдов, идущих через пустыню. Когда это не помогло, он стал считать суммарное количество цифр S во всех целых числах от M до N включительно при различных значениях M и N .

Вася заинтересовался обратной задачей: как при заданном количестве цифр S определить соответствующий диапазон (M, N) ? Он быстро понял, что достаточно взять единственное число P из S цифр и положить $M = N = P$. Но это было совсем неинтересно.

Далее Вася задумался о том, как минимизировать M , не превышая некоторого предела L . Сон окончательно пропал. Чтобы Вася спал спокойно, необходимо написать программу для решения этой задачи.

Формат входных данных

В единственной строке входного файла находятся целые числа S и L ($1 \leq S \leq 10^{18}$; $1 \leq L \leq 10^4$).

Формат выходных данных

В единственную строку выходного файла вывести через пробел значения M и N ($1 \leq M \leq N$; $M \leq L$). Если ответов несколько, вывести тот, для которого величина M минимальна. Если решения нет, вывести *No*.

Примеры:

№	input.txt	output.txt
1	2 5	1 2
2	12 20	2 11
3	10 1	No

Задача D. Автогонка

Имя входного файла: input.txt

Имя выходного файла: output.txt

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Круг кольцевой автогонки имеет длину S километров. Движение происходит только в одном направлении. Автогонщик Веттель отработывает на тренировке остановку и дозаправку бензином.

На расстояниях X_1, X_2, \dots, X_N от старта расставлены N канистр бензина. Бензобак в состоянии вместить содержимое всех канистр. Болид Веттеля с пустым бензобаком доставляется к одной из канистр. После заправки бензином из этой канистры болид начинает движение по кольцу, останавливаясь и заправляясь дополнительно у каждой встреченной непустой канистры. Движение по кольцу продолжается до тех пор, пока бензобак не опустеет.

Известно, что бензина из i -й канистры, находящейся в точке X_i , хватает на C_i километров. Требуется выбрать стартовую канистру так, чтобы Веттель смог проехать максимальное расстояние.

Формат входных данных

В первой строке входного файла заданы через пробел значения N и S ($1 \leq N \leq 2 \times 10^5$; $1 \leq S \leq 10^9$). Во второй строке находятся через пробел значения $X_1 < X_2 < \dots < X_N$ ($0 \leq X_i < S$). В третьей строке указаны через пробел значения C_1, C_2, \dots, C_N ($1 \leq C_i \leq 10^5$). Все числа целые.

Формат выходных данных

В единственной строке выходного файла вывести расстояние, которое пройдет болид при движении от найденной стартовой канистры.

Примеры:

№	input.txt	output.txt	Комментарий
1	3 10 1 3 6 3 4 1	8	
2	3 10 1 5 9 3 1 3	7	При движении от канистры 3 Веттель по кругу добирается до канистры 1. В бензобаке остается бензин на 1 км, а после дозаправки его хватит на 4 км. Этого бензина достаточно для того, чтобы доехать до канистры 2. Бак оказывается пустым, но после дозаправки из канистры 2 Веттель может проехать еще 1 км. Общая протяженность пути составляет 7 км
3	3 10 2 5 9 2 7 1	10	

Задача Е. Ремонт дорог

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В тридесятом государстве имеется N городов ($3 \leq N \leq 300$). Некоторые из них связаны асфальтовыми дорогами. Между двумя городами имеется не более одной дороги. Города пронумерованы от 1 до N . По каждой дороге возможно движение в обоих направлениях. Из каждого города имеется путь в любой другой город напрямую либо через промежуточные города.

Решено отремонтировать часть дорог, покрыв их супербетоном марки «Патриот». Для каждой пары городов должен быть обеспечен хотя бы один кратчайший путь по дорогам с бетонным покрытием, связывающий эти города. При этом общая длина ремонтируемых дорог должна быть минимальной. Требуется по заданным длинам кратчайших путей между всеми городами определить дороги для предстоящего ремонта.

Формат входных данных

Первая строка входного файла содержит значение N . Далее следует $N-1$ строка описания кратчайших расстояний между городами. Каждая i -я строка содержит $N-i$ значений C_{ij} ($1 \leq C_{ij} \leq 10^5$; $j = i+1, i+2, \dots, N$), где C_{ij} – кратчайшее расстояние от i -го до j -го города.

Формат выходных данных

Если решения нет, вывести в первой строке *No*. В противном случае в первой строке выводятся количество новых дорог M и их общая длина L . Следующие M строк содержат по три числа: номера двух городов, определяющих дорогу, и ее длина.

Дороги выводятся в порядке возрастания первого номера в паре, а при равных первых номерах – в порядке возрастания второго номера. Первый номер в паре должен быть меньше второго. Если имеется несколько решений, вывести любое из них.

Примеры:

№	input.txt	output.txt
1	4	4 9
	2 3 4	1 2 2
	1 3	1 4 4
	2	2 3 1
		3 4 2

№	input.txt	output.txt
2	4	3 4
	1 2 4	1 2 1
	1 3	2 3 1
	2	3 4 2
3	3	2 3
	1 3	1 2 1
	2	2 3 2

Задача F. Найти кольцо

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Прямоугольная комната размером $M \times N$ метров с мебелью покрыта квадратными паркетными плитами со стороной один метр. В углу комнаты стоит тяжелый сейф, занимая всю площадь плиты. Под сейф закатилось кольцо с бриллиантом, поэтому решено полностью освободить плиту под сейфом. Для этого нужно переместить сейф на одну из соседних плит путем кантования – поворота в плоскости пола на 90° вокруг одного из своих углов. Соседними считаются плиты, имеющие общую сторону.

Некоторые плиты комнаты заняты стульями. Каждый стул может быть переставлен на свободную соседнюю с ним плиту. В комнате может быть и другая мебель. Она также занимает определенные плиты, но в отличие от стульев не может перемещаться.

При повороте сейфа никакая его точка не должна пересекать внутреннюю часть такой плиты, на которой стоит стул или другая мебель.

Матрица $M \times N$ определяет начальное положение мебели. Каждый элемент матрицы описывает одну паркетную плиту. Сейф находится в левом верхнем углу. Плите с сейфом соответствует символ «#», свободной плите паркета – символ «.» (точка), плите со стулом – символ «@», плите с другой мебелью – символ «*». Поворот сейфа выполняется за одну минуту, а перестановка стула – за одну секунду. Найти минимальное время в секундах, за которое сейф может быть перемещен.

Формат входных данных

В первой строке содержатся числа M и N ($2 \leq M, N \leq 500$), разделенные пробелами. В следующих M строках находится матрица, описывающая комнату, по N подряд идущих символов в строке.

Формат выходных данных

Вывести наименьшее число секунд, необходимых для освобождения плиты под сейфом. Если плиту освободить невозможно, вывести *No*.

Примеры:

№	input.txt	output.txt
1	4 4 #... @.@. ...@	60

№	input.txt	output.txt
2	4 3 #. * . @ . @ * * @ . @	61

№	input.txt	output.txt
3	4 2 #. . @ @ * @ @	No

Задача G. Полеты со скидкой

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Самолеты авиакомпании «Себе в убыток» летают в N городов ($2 \leq N \leq 100$) и обслуживают M рейсов. Города пронумерованы от 1 до N . Каждый рейс связывает два разных города. Между парой городов может быть не более чем по одному рейсу в каждом направлении.

Билет на каждый рейс имеет определенную стоимость. Компания продает единый билет на некоторый маршрут, предусматривающий пересадки с одного рейса на другой. В этом случае если маршрут имеет не менее K пересадок ($1 \leq K < N$), то после K -й пересадки стоимость на каждый рейс маршрута снижается вдвое. Маршрут может иметь повторяющиеся города и рейсы.

Требуется рассчитать минимальную стоимость перелета из столицы, имеющей номер 1, в другие города.

Формат входных данных

Первая строка входного файла содержит целые положительные числа N , M и K через пробел. Далее следуют M строк описания рейсов по три целых положительных числа A , B и C в строке ($1 \leq A, B \leq N$, $2 \leq C \leq 10\,000$). Первые два числа задают номера городов, а третье – стоимость перелета. Известно, что стоимость каждого рейса выражается четным числом.

Формат выходных данных

В i -й строке выводятся минимальная стоимость перелета от столицы до $i+1$ -го города. Если нет ни одного пути, в соответствующей строке выводится 0. Таким образом выходной файл состоит из $N-1$ строки.

Пример:

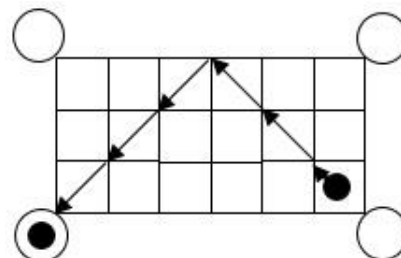
№	input.txt	output.txt	Комментарий
1	3 3 2 1 2 4 2 1 4 2 3 20	4 20	Для полета в город 3 выгоднее всего слетать в город 2 и вернуться в столицу. Следующий повторный перелет в город 2 происходит после двух пересадок и обходится вдвое дешевле, как и последний перелет в город 3.

Задача Н. Бильярд

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Бильярдный стол расчерчен на квадратные клетки и имеет размеры $M \times N$ клеток. В углах стола находятся четыре лузы для шаров. В центрах некоторых клеток находятся шары. По определенному шару, называемому битком, наносится удар, в результате которого шар может катиться в одном из четырех направлений вдоль диагоналей клетки. Если этот шар достигает края стола, то отражается и продолжает движение, а если попадает в угол стола, то сваливается в лузу и игра заканчивается. Если биток не встречает на своем пути других шаров и не сваливается в лузу, то он проходит ровно K клеток, не считая начальной, но включая конечную клетку либо лузу. Если же биток достигает клетки с другим шаром, то встреченный шар удаляется, а биток занимает его место и используется для последующих ударов. По заданному расположению шаров найти наименьшее количество ударов, необходимое для снятия всех шаров, включая биток.

На рисунке показано возможное движение битка на свободном столе из клетки с координатами (3, 6). Он попадает в клетку (3, 1), проходя 5 клеток, а в левую нижнюю лузу, проходя 6 клеток.



Формат входных данных

Первая строка содержит целые числа M , N и K ($1 \leq M, N \leq 15$; $1 \leq K \leq 10$) через пробел. Каждая i -я строка из следующих M строк содержит N символов C_{ij} ($1 \leq j \leq N$). Свободные клетки стола помечаются символами «.», клетки с неподвижными шарами символами «#», а клетка с битком – символом «&». Общее количество шаров, включая биток, может быть от 1 до 15.

Формат выходных данных

В единственной строке выводится минимальное количество ударов, необходимое для снятия всех шаров. Если это невозможно, вывести *No*.

Примеры:

№	input.txt	output.txt	Комментарий
1	2 3 2 #.. &.#	3	Биток из клетки (2, 1) после первого удара вверх и налево вытесняет шар с клетки (1, 1). Вторым ударом вниз и направо шар попадает в клетку (2, 3), вытесняя еще один шар. Наконец, третьим ходом вниз и направо он сваливается в лузу
2	3 3 5 .#. .&. ...	No	Любой удар приводит к попаданию битка в лузу. При этом на столе остается шар в клетке (1, 2).