

Необходимо сделать на Рабочем столе папку с именем по шаблону «Группа_Фамилия_Программирование» и все создаваемые в ходе работы файлы сохранять в ней.

Задача 1 «Простые числа» (10 баллов)

Необходимо вывести все простые числа от M до N включительно.

Входные данные:

Входной файл **Z1.txt** содержит два натуральных числа M и N , разделенных пробелом ($2 \leq M \leq N \leq 10^6$).

Выходные данные:

Выведите все простые числа от M до N в порядке возрастания в строку, через пробел. Если таких чисел нет, то следует вывести «Absent».

Примеры:

№	Входные данные	Выходные данные
1	2 5	2 3 5
2	4 4	Absent
3	150 167	151 157 163 167

Задача 2 «Попрыгунчик» (15 баллов)

На вершине лесенки, содержащей N ступенек, находится попрыгунчик, который начинает прыгать по ним вниз, к основанию. Он может прыгнуть на следующую ступеньку, на ступеньку через одну или через две. То есть, если попрыгунчик находится на 8-ой ступеньке, то он может переместиться на 5-ую, 6-ую или 7-ую.

Требуется написать программу, которая определит число всевозможных «маршрутов» попрыгунчика с вершины на землю.

Входные данные:

Входной файл **Z2.txt** содержит число N ($0 < N \leq 70$).

Выходные данные:

Выведите искомое число всевозможных «маршрутов» попрыгунчика с вершины на землю.

Примеры:

№	Входные данные	Выходные данные
1	1	1
2	4	7

Задача 3 «Выставка эксцентричного художника» (20 баллов)

В одной из художественных галерей города неким известным и эксцентричным художником была организована необычная выставка-лабиринт. Пространство галереи переоборудовали таким образом, что оно стало состоять из N комнат, соединенных M коридорами. По задумке художника каждая комната – это точка выбора, их сделали небольшими и картин вешать не стали. Куда более ценным художник считал именно приобретённый опыт и пройденный путь – коридор. Каждый коридор покрасили в *один из ста* цветов и украсили картинами с преобладанием этого оттенка.

При этом от каждой комнаты отходит не более одного коридора каждого цвета. Две комнаты могут быть соединены любым количеством коридоров.

Посетитель галереи, купивший билет, оказывается в комнате номер один. К билету прилагается описание пути, составленное самим маэстро, по которому посетитель может добраться до выхода из выставки-лабиринта. Это описание представляет собой последовательность цветов $C_1 \dots C_k$. Пользоваться ей следует так: находясь в комнате, надо посмотреть на очередной цвет в этой последовательности, выбрать коридор такого цвета и пойти по нему. Если из комнаты нельзя пойти по коридору соответствующего цвета, то посетителю приходится дальше самому выбирать, куда идти.

В последнее время в администрацию художественной галереи стали часто поступать жалобы от заблудившихся людей. Маэстро, приняв эффектную и таинственную позу, заявил, что ошибки на жизненном пути делают его более насыщенным, и от дальнейших комментариев отказался. В связи с этим, возникла необходимость написания программы, проверяющей правильность описания пути, и, в случае его корректности, сообщаящей номер комнаты, в которую ведет путь. Описание пути некорректно, если возникает ситуация, когда из комнаты нельзя пойти по коридору соответствующего цвета.

Входные данные:

Первая строка входного файла **Z3.txt** содержит два целых числа n ($1 \leq n \leq 10000$) и m ($1 \leq m \leq 10000$) - соответственно количество комнат и коридоров в выставке-лабиринте. Следующие m строк содержат описания коридоров. Каждое описание содержит три числа u ($1 \leq u \leq n$), v ($1 \leq v \leq n$), c ($1 \leq c \leq 100$) - соответственно номера комнат, соединенных этим коридором, и цвет коридора. Следующая, $(m+2)$ -ая

строка входного файла содержит длину описания пути - целое число k ($0 \leq k \leq 10000$). Последняя строка входного файла содержит k целых чисел, разделенных пробелами - описание пути.

Выходные данные:

Выведите строку INCORRECT, если описание пути некорректно, иначе выведите номер комнаты, в которую ведет описанный путь. Обратите внимание, что путь начинается в комнате номер один.

Примеры:

№	Входные данные	Выходные данные
1	3 2 1 2 10 1 3 5 5 10 10 10 10 5	3
2	3 2 1 2 10 2 3 5 5 5 10 10 10 10	INCORRECT

Задача 4 «Силовые поля» (25 баллов)

В физико-биологической лаборатории исследуют воздействие излучения на растения при облучении через силовые поля.

Экспериментальная установка содержит квадратную платформу размером 109×109 , заполненную плодородной почвой. Над платформой установлен источник излучения. Между источником излучения и платформой можно включать n силовых полей.

Генератор силовых полей установлен над точкой $(0, 0)$. При этом i -е силовое поле представляет собой прямоугольник со сторонами, параллельными границам платформы и координатами двух противоположных углов $(0, 0)$ и (x_i, y_i) .

В эксперименте планируется изучать воздействие излучения на растения при облучении через k силовых полей. Из заданных n полей необходимо выбрать k полей для эксперимента. Ученые хотят выбрать силовые поля таким образом, чтобы площадь участка платформы, над которой находятся все k выбранных силовых полей, была максимальна.

Требуется написать программу, которая по заданным целым числам n , k и описанию n силовых полей определяет, какие k силовых полей необходимо выбрать для эксперимента, чтобы площадь участка, покрытого всеми k силовыми полями, была максимальна, и выводит это значение.

Входные данные:

Первая строка входного файла **Z4.txt** содержит целые числа n и k ($1 \leq k \leq n \leq 200\,000$) – общее количество силовых полей и количество силовых полей, которые необходимо выбрать для эксперимента.

Последующие n строк содержат по два целых числа x_i, y_i ($1 \leq x_i, y_i \leq 109$) – координаты дальнего от начала координат угла прямоугольного участка i -го силового поля.

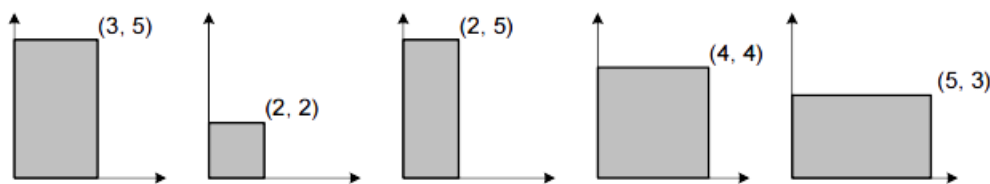
Выходные данные:

Выведите одно целое число: максимальную площадь искомого участка.

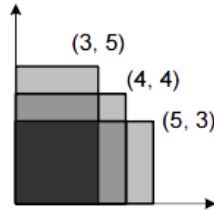
Примеры:

№	Входные данные	Выходные данные
1	5 3 3 5 2 2 2 5 4 4 5 3	9

На рисунке показаны пять силовых полей, заданных во входном файле.



Оптимальный способ выбрать из них три поля для эксперимента:



Задача 5 «Каникулы Гарри Поттера» (30 баллов)

Железная дорога от платформы девять и три четверти представляет собой прямую, вдоль которой расположены n станций. Будем называть участок железной дороги от некоторой станции до следующей перегонном.

Поезд следует от станции 1 до станции n , делая остановку на каждой станции. В поезде k мест, пронумерованных от 1 до k . На поезд продаются билеты, каждый билет характеризуется тремя числами: s , t и a . Такой билет позволяет проехать от станции s до станции t на месте a .

Гарри планирует в один из дней летних каникул проехать на поезде от одной станции до другой. Он выяснил, что на поезд в этот день уже продано m билетов, и возможно уже нет мест, свободных на всех перегонах между интересующими его станциями. Билет от одной станции до другой на определенное место можно купить, только если это место свободно на всех перегонах между этими станциями.

Гарри сообразил, что иногда все равно можно проехать от одной станции до другой, купив несколько билетов и пересаживаясь с одного места на другое на некоторых промежуточных станциях. Разумеется, пересаживаться с места на место неудобно, поэтому экономный Поттер хочет купить минимальное количество билетов, чтобы на каждом перегоне у него было свое место.

Гарри еще не решил, от какой станции и до какой он поедет. Он записал q вариантов поездки, и для каждого из них хочет узнать, какое минимальное число билетов ему придется купить, если он выберет этот вариант.

Требуется написать программу, которая по заданному описанию уже проданных билетов и вариантов поездки Гарри Поттера определяет для каждого варианта, какое минимальное количество билетов необходимо купить, чтобы совершить такую поездку.

Входные данные:

Первая строка входного файла **Z5.txt** содержит числа n , m и k ($2 \leq n \leq 100$, $0 \leq m \leq 100$, $1 \leq k \leq 100$) – количество станций, количество уже проданных билетов и количество мест в поезде. Последующие m строк содержат информацию о проданных билетах. Каждая строка содержит три числа: s_i , t_i и a_i – номер станции, от которой куплен билет, номер станции, до которой куплен билет, и номер места, на которое куплен билет ($1 \leq s_i < t_i \leq n$, $1 \leq a_i \leq k$). Гарантируется, что все билеты куплены таким образом, что ни на каком перегоне ни на какое место нет более одного билета.

Далее идет строка, которая содержит число q ($1 \leq q \leq 100$). Последующие q строк содержат описания вариантов поездки. Каждая строка содержит два числа: f_j , d_j – номер станции, от которой Гарри хочет поехать в этом варианте, и номер станции, до которой он хочет поехать ($1 \leq f_j < d_j \leq n$).

Выходные данные:

Выведите q чисел: для каждого варианта поездки требуется вывести минимальное количество билетов, которое необходимо купить Гарри Поттеру, чтобы совершить соответствующую поездку. Если поездку совершить невозможно, то для этого варианта требуется вывести -1 .

Примеры:

№	Входные данные	Выходные данные
1	5 4 3	-1
	1 4 1	2
	2 5 3	1
	2 3 2	
	4 5 2	
	3	
	1 5	
	3 5	
	4 5	