

Время выполнения заданий: 240 минут

Максимальное количество баллов - 100

1. (10 баллов) Один из простейших способов кодирования – это поменять местами определённые биты в представлении числа. Предположим, что нужно поменять местами два старших и два младших бита в байте 10100111. Результат кодирования – байт 11100110. В байтах хранятся целые беззнаковые числа. Таким образом, содержимое в десятичном представлении изменилось с 167 на 230. Меняя количество переставляемых бит, а также количество байтов для представления чисел, можно получить разные методы кодирования. Входные данные – набор трёх чисел: число, которое нужно закодировать; количество байтов, в которых хранится первое число; количество старших и младших битов, которые нужно поменять местами. Выходные данные – закодированное число.

Пример.

Вход	Выход	Примечание
167 1 2	230	10100111 → 11100110
32780 2 1	13	1000000000001100 → 0000000000001101

Вам представляется десять наборов из трёх чисел, необходимо записать в ответе десять чисел (разделитель – пробел), полученных после кодирования описанным способом.

7 1 4
27 1 4
51 1 2
137 1 2
183 1 3
11 2 1
7 2 2
143 2 8
40967 2 3
61129 2 4

2. (10 баллов) При хранении данных одна из основных задач — соблюдение баланса между расходами на количество дисков и надёжностью записи. Одним из компромиссных по надёжности и стоимости хранения данных является RAID-4 – избыточный массив независимых дисков с выделенным диском для хранения блоков с битами чётности.

Наш RAID-4 состоит из пяти дисков, на четырёх из которых содержится информация, а на пятом – блоки контрольных битов чётности. При записи четырёх двухбайтных блоков (по блоку на каждый из четырёх дисков) вычисляется контрольный блок чётности, составленный из контрольных битов. Для каждого из 16 битов вычисляется сумма значений этих битов во всех блоках данных, при этом значение контрольного бита выбирается так, чтобы сумма значений во всех разрядах, включая контрольный, была чётной.

Например, у нас есть два основных диска и на них записывается байты 1001001011100111 и 0111011100011001. Тогда значение контрольного байта равно 111001011111110.

Один из четырёх основных дисков в RAID-4 вышел из строя. Известны значения блоков (беззнаковые двухбайтовые числа) в трёх оставшихся дисках и значение байта на контрольном диске. Все числа приведены в десятичной системе счисления.

– значения на первых трёх дисках: 17257, 17257, 17257, контрольный байт: 17257;

Информатика

- значения на первых трёх дисках: 7969, 7969, 7969, контрольный байт: 0;
- значения на первых трёх дисках: 65326, 65336, 65528, контрольный байт: 65370;
- значения на первых трёх дисках: 65535, 0, 150, контрольный байт 9604;
- значения на первых трёх дисках: 137, 232, 23, контрольный байт 212.

Какой блок был записан на сломавшемся диске? Ответы также следует привести в десятичной системе счисления. Ответы разделять пробелом.

3. (15 баллов) Перед началом игры «Морской бой» игрок заполняет игровое поле из $N \times N$ клеток «кораблями» (прямоугольниками $m \times n$ клеток). Размер корабля – количество клеток, которое занимает корабль (т.е. mn). Корабли могут иметь только прямоугольную форму (изгибы, например, в виде буквы Г, не допускаются). Корабли не имеют общих и соприкасающихся (сторонами или вершинами) клеток. (Проверять правильность заполнения игрового поля не нужно.)

Напишите программу, которая получает на вход заполненное игроком игровое поле. В первой строке: N – длина стороны игрового поля в клетках ($0 < N < 101$); далее построчно – обозначения клеток поля (0 – клетка свободна, 1 – клетка занята). В ответе должно быть целое число: средний размер кораблей. (Дробную часть числа отбрасывать.)

Пример.

Вход	Выход
4 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0	1
7 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0	4

4. (20 баллов) Трёхмерные шахматы представляют собой расширение традиционных шахмат. Шахматную доску заменяет куб с ребром N . Каждое «поле» (кубик) имеет координаты – тройку чисел от 1 до N . Одно из угловых «полей» (например, левый нижний передний кубик) считается началом отсчёта: он имеет координаты: 1 1 1.

Для фигур определены следующие возможные ходы.

Король может ходить на любое соседнее «поле», т.е. на любой кубик внутри куба $3 \times 3 \times 3$, где в центре в данный момент находится фигура короля. При этом король бьёт фигуру противника, стоящую на «поле», куда он перемещается.

Ладья может ходить на любое количество «полей», если на её пути не стоят другие фигуры, в любом из шести направлений: вперёд, назад, влево, вправо, вверх, вниз. При этом ладья бьёт фигуру противника, стоящую на «поле», куда она перемещается.

В описываемый игровой момент у белых осталось три фигуры: король и две ладьи, а у черных K фигур. Их координаты вводятся в программу, которая должна определить, сколько черных фигур НЕ находятся под ударом белых.

Порядок ввода:

N – длина ребра игрового куба ($0 < N < 101$);

Три строки по три целых числа – координаты короля и двух ладей белых, соответственно;

K – количество чёрных фигур ($0 < K < 101$);

K строк: в каждой строке три целых числа – координаты чёрных фигур.

Пример:

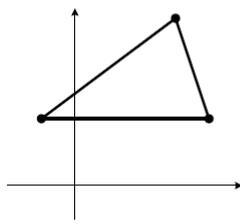
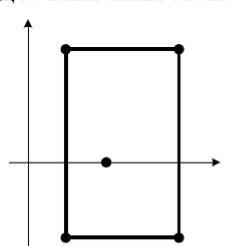
Информатика

Вход	Выход	Примечание
10 8 2 5 1 1 1 9 1 1 4 3 1 1 9 1 5 9 9 10 1 1 5	3	король 1-я ладья 2-я ладья бьётся двумя ладьями бьётся королем и 2-й ладьей не бьётся ни одной фигурой бьётся 1-й ладьей

5. (20 баллов) В корпоративном тренинге соревнуются несколько команд. Каждой команде выделяется поляна, на которой за ограниченное время надо случайным образом установить как можно больше колышков. При этом колышки можно устанавливать только в гнезда, находящиеся в узлах некой координатной сетки. После того, как отведённое для игры время истекло, на колышки каждой команды натягивают ленту, огораживая участок максимальной возможной площади. При этом лента может прилегать не к каждому колышку. Выигрывает команда, площадь участка которой больше. Необходимо написать программу, которая помогает организаторам вычислять площадь огороженного участка.

На вход подаётся целое число N – количество установленных командой колышков ($0 < N < 101$), а затем подаётся N пар целых чисел – координат колышков. На выходе одно число: площадь огороженного участка – вещественное число с точностью до сотых (т.е. до двух знаков после десятичного разделителя).

Пример.

Вход	Выход	Примечание
3 -1 2 4 2 3 5	7.50	треугольник 
5 1 -2 2 0 4 -2 4 3 1 3	15.00	Прямоугольник (колышек с координатами (2, 0) ленточкой не обвязан для достижения максимальной площади) 

6. (25 баллов) Команда студентов ВШЭ прошла на финал чемпионат мира по программированию, который должен проходить в г. Марракеш. Пришло время покупать авиабилеты. Спонсор чемпионата мира, авиакомпания ACM-Airlines, подарила каждому участнику команды сертификат на один бесплатный прямой перелёт туда и обратно.

К сожалению, у ACM-Airlines нет прямого рейса Москва-Марракеш, поэтому участникам придётся лететь с пересадками и оплачивать все сегменты перелёта, кроме одного, который можно получить бесплатно по сертификату. Все рейсы ACM-Airlines летают в оба направления и продаются только парой туда и обратно. Некоторые пары

Информатика

городов не соединены прямыми рейсами, но с пересадками можно добраться от любого города до любого другого. Между парой городов у ACM-Airlines не бывает больше одного рейса.

Студенты написали скрипт, получающий данные о ценах на билеты, и ушли тренироваться к финалу, а задача поиска самого дешёвого перелёта осталась Вам.

В данных о перелётах города занумерованы от 1 до N , где 1-й город – Москва, а N -й – Марракеш.

Входные данные. В первой строке ввода задано два целых числа: N – количество городов и M – количество прямых рейсов между ними. В следующих M строках записано по три натуральных числа: A_i и B_i – номера городов, между которыми существует прямой перелёт, и C_i – стоимость перелёта (суммарная стоимость за перелёт туда и обратно). Гарантируется, что не существует рейса из города 1 в город N .

Выходные данные. В первой строке выведите два натуральных числа: Sum – минимальную стоимость, Len – количество сегментов перелёта. Во второй строке выведите номера городов в том порядке, в котором их следует посещать на пути из Москвы в Марракеш (их количество равно $Len + 1$; каждая соседняя пара городов в этом списке обозначает, что между ними был прямой перелёт.). В третьей строке выведите пару чисел – номера городов, соединённых прямым рейсом, перелёт между которыми нужно оплатить сертификатом. Города в паре можно выводить в любом порядке. Если правильных ответов несколько – выведите любой из них.

Пример.

Вход	Выход
4 4	2 2
2 1 10	1 2 4
3 4 5	1 2
2 3 2	
1 4 5	