

Муниципальный этап всероссийской олимпиады школьников по информатике

Задания для 7–8 классов

В задачах 1–4 ответ записывается в виде числа, строки, некоторого многострочного текста в соответствии с форматом, описанным в условии задачи. Во время тура в задачах 1–4 производится проверка ответа на соответствие формату, описанному в условии задачи. Если формат записи ответа соответствует условию задачи, решение получает статус «Принято на проверку», иначе решение получает статус «Неправильный формат ответа», в этом случае в протоколе проверки указаны причины этого. После окончания олимпиады будет проверено и оценено последнее принятое на проверку решение по каждой задаче 1–4.

Задача 1. Римские числа

Один древнеримский торговец брал несколько раз ссуду в древнеримском банке. Каждый раз банкир записывал размер выданной ссуды на листе пергамента, используя римские числа. Но ввиду дороговизны пергамента запись производилась плотно и все числа оказались записанными подряд, без разделителей. Когда торговец пришёл возвращать ссуду, оказалось, что невозможно установить разбиение записи на отдельные числа.

Например, если на пергаменте записана строка «XIIIV», её можно разбить на римские числа разными способами, например, $XI + IV = 11 + 4 = 15$ или $XII + V = 12 + 5 = 17$, возможны и другие варианты разбиения.

Торговец хочет вернуть как можно меньше денег, поэтому он хочет разбить строку цифр на корректные римские числа так, чтобы сумма всех чисел была как можно меньше.

Вам необходимо решить задачу для следующих пяти строк.

VXIVVIX
CXLVLXC
XIXCXIXL
XCMLXCD
VLMIXCD

Запишите в ответе пять чисел (арабскими цифрами в десятичной системе счисления), каждое число в отдельной строке – ответы для данных пяти строк точно в таком же порядке. Например, для строки «XIIIV» ответом будет число 15. Если вы не можете дать ответ для какой-либо строки, напишите в ответе для этой строки число 0.

Правила записи римских чисел

Цифра	Тысячи	Сотни	Десятки	Единицы
1	M	C	X	I
2	MM	CC	XX	II
3	MMM	CCC	XXX	III
4		CD	XL	IV
5		D	L	V
6		DC	LX	VI
7		DCC	LXX	VII
8		DCCC	LXXX	VIII
9		CM	XC	IX

Римскими цифрами можно записать целые числа от 1 до 3999. Число представляется в виде суммы тысяч, сотен, десятков и единиц. Далее из следующей таблицы берётся по

одному элементу, соответствующему тысячам, сотням, десяткам, единицам ровно в таком порядке.

Если число тысяч, сотен, десятков, единиц равно 0, то из соответствующего столбца ничего не берётся. Например, число 1990 записывается, как $1000 + 900 + 90 = \text{MCMXC}$.

Задача 2. Номера домов

На школьном этапе олимпиады была задача по программированию, в которой нужно было посчитать расстояние между двумя домами на улице. В этой задаче человек идёт по улице и складывает номера домов, находящихся с одной стороны улицы (то есть все номера домов имеют одинаковую чётность). Он начал с дома номер a и закончил домом номер b . Определите сумму всех номеров домов, которые он сложил (он складывал все числа одной чётности, начиная с числа a до числа b включительно, числа a и b одной чётности, $a \leq b$). Например, если $a = 3$ и $b = 7$, то он сложил числа $3 + 5 + 7$ и получил в ответе 15.

Ответом на эту задачу является некоторое выражение, которое может содержать целые числа, переменные a и b (записываемые английскими буквами), операции сложения (обозначаются «+»), вычитания (обозначаются «-»), умножения (обозначаются «*»), деления (обозначаются «/») и круглые скобки для изменения порядка действий. Запись вида « $2a$ » для обозначения произведения числа 2 и переменной a неверная, нужно писать « $2 * a$ ».

Если ваше решение работает только в одном случае: для чётных или нечётных чисел a и b , вы всё равно получите неполный балл за эту задачу.

Пример правильной формы записи ответа.

$$b / 2 + (a * b - b) * 2$$

Задача 3. Сортировка

На уроке информатики учитель показывает фокус с сортировкой чисел. На столе стоят 4 шкатулки, обозначенные слева направо буквами А, В, С, D. Ученик кладёт в каждую шкатулку лист бумаги, на котором написано число от 1 до 4, все числа различные. Учитель не знает, какое число лежит в каждой шкатулке.

Затем учитель записывает последовательность команд и отдаёт её ученику. Каждая команда содержит в себе две разные буквы. Ученик последовательно выполняет эти команды. Он открывает две шкатулки, указанные в команде, достаёт из них листки с числами, сравнивает эти числа и затем листок с меньшим из чисел кладёт в левую открытую шкатулку (шкатулку с меньшей буквой), а листок с большим из чисел – в правую открытую шкатулку (шкатулку с большим номером).

Например, пусть в шкатулках лежат числа 2, 1, 4, 3.

Шкатулка	A	B	C	D
Число внутри неё	2	1	4	3

Учитель написал на листке следующие команды.

AB
BC

Ученик открывает шкатулки А и В, достаёт из них числа 2 и 1. В шкатулку А кладёт число 1, в шкатулку В – число 2.

Шкатулка	A	B	C	D
Число внутри неё	1	2	4	3

Затем ученик открывает шкатулки В и С, достаёт из них числа 2 и 4. В шкатулку В кладёт число 2, в шкатулку С – число 4. Содержимое шкатулок не изменилось.

Фокус заключается в том, что в результате исполнения алгоритма числа в шкатулках оказываются упорядочены независимо от того, какие числа первоначально лежали в шкатулках. То есть в шкатулке А оказывается число 1, в шкатулке В – число 2, в шкатулке С – число 3, в шкатулке D – число 4.

Помогите выполнить этот фокус: придумайте такую последовательность команд, которая упорядочивает числа в шкатулках при любом первоначальном расположении чисел.

Ответ на эту задачу нужно записать в виде нескольких строк. Каждая строка содержит две различные буквы из множества A, B, C, D. Чем короче будет ваша последовательность команд, тем больше баллов вы получите.

Задача 4. Двоичное кодирование

При двоичном кодировании буквам сопоставляются последовательности из символов «0» или «1». Например, пусть дан следующий код:

Код буквы «К»:	0
Код буквы «Н»:	10
Код буквы «О»:	11

Тогда слово «ОКНО» кодируется в виде последовательности «1101011»: коды всех букв записываются последовательно, один за другим.

Особую роль в теории кодирования играют *префиксные коды*. Код называется префиксным, если код ни одной из букв не является началом (префиксом) кода какой-либо другой буквы. Код, приведённый выше, является префиксным.

Префиксные коды являются «хорошими», потому что обеспечивают однозначность декодирования, то есть нельзя записать две различные последовательности букв, которые будут закодированы одинаковым способом.

Приведём пример кода, не являющегося префиксным.

Код буквы «К»:	0
Код буквы «Н»:	10
Код буквы «О»:	1

В этом примере код буквы «О» (1) является началом кода буквы «Н» (10).

Если использовать такой код, то кодом слова «ОКНО» будет «10101», но точно так же будет кодироваться и слово «НОКО». Более простой пример неоднозначного кодирования: последовательности букв «Н» и «ОК» кодируются одинаково: «10».

Кодовые слова могут иметь различную длину, при этом длина закодированного сообщения может оказаться различной. Например, рассмотрим следующий код.

Код буквы «К»:	00
Код буквы «Н»:	01
Код буквы «О»:	1

При использовании такого кода слово «ОКНО» будет закодировано строкой «100011», то есть будет короче, чем в ранее рассмотренном примере. Идея неравномерного кодирования (кодирования разных символов при помощи кодовых слов переменной длины) используется, например, при построении архиваторов.

Задание

Рассмотрим следующую строку:

КЛОУН УКОЛОЛ АКУЛУ ОКОЛО КОЛОКОЛА

В этой строке встречаются следующие буквы: А, К, Л, Н, О, У. Пробелы учитывать не будем, считаем, что строка не содержит пробелов.

Вам нужно придумать *двоичный префиксный* код для букв А, К, Л, Н, О, У, такой, чтобы при использовании данного кода приведённая выше строка кодировалась бы как можно более короткой последовательностью символов 0 и 1.

Ответ нужно записать в форме шести строк, являющихся кодами букв А, К, Л, Н, О, У, именно в таком порядке. Каждая строка может содержать только символы 0 и 1. Построенный код должен быть префиксным. При использовании данного кода строка «КЛОУН УКОЛОЛ АКУЛУ ОКОЛО КОЛОКОЛА» должна кодироваться строкой из символов

0 и 1 минимальной длины. Чем короче будет полученная при кодировании строка, тем больше баллов вы получите.

Например, если требуется закодировать строку «ОКНО» и в ответе нужно записать коды букв К, Н, О, то для получения закодированной строки минимальной длины ответ следует записать в таком виде:

00
01
1

Задачи по программированию

Решением задач 5–7 является программа, написанная на одном из языков программирования. Ограничение по времени работы программы в задачах 5–7 – 1 секунда. Ограничение по памяти – 256 МВ.

Во всех задачах целые числа во входных и выходных данных записываются только цифрами (т.е. недопустимо использование записи 1000000.0 или 1e6 вместо числа 1000000). Каждое число во входных данных записано в отдельной строке.

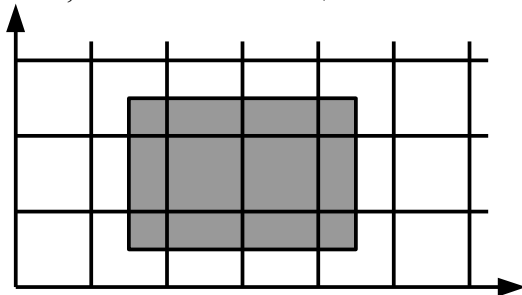
Задача 5. Плитка

Стена покрыта квадратной плиткой со стороной M см. На стену повесили картину, известны её ширина и высота. Определите максимальное количество плиток, которые могли оказаться частично или полностью закрыты картиной.

Первая строка входных данных содержит число M – сторону плитки. Вторая и третья строки содержат числа W и H – ширину и высоту картины. Все числа целые положительные, не превосходящие 2×10^9 .

Программа должна вывести одно число – максимальное количество плиток, которые могли оказаться частично или полностью закрыты картиной.

Плитка считается закрытой картиной, если пересечение картины и плитки имеет ненулевую площадь, то есть касание картины и плитки не считается закрытием плитки. Стороны картины должны быть параллельны сторонам плитки. Углы картины могут находиться в произвольных точках, необязательно с целочисленными координатами.



Пример входных и выходных данных

Ввод	Вывод	Примечание
10 30 20	12	Пример соответствует картинке. Сторона плитки (сторона клетки на картинке) $M = 10$. Картина имеет ширину 30 см и высоту 20 см. Картина полностью или частично закрывает 12 плиток.

Система оценивания

Решение, правильно работающее только для случаев, когда все входные числа не превосходят 100, будет оцениваться в 6 баллов.

Задача 6. Лифт в бизнес-центре

Бизнес-центр представляет собой N -этажное здание, этажи пронумерованы от 1 до N снизу вверх. На каждом этаже работает ровно один сотрудник. Все сотрудники утром приезжают на парковку, которая расположена в подвальном помещении, на один этаж ниже первого. Бизнес-центр оборудован лифтом, который вмещает неограниченное число людей, но вредный лифтёр сегодня готов отвезти всех сотрудников только на один какой-то этаж. С этого этажа сотрудники расходятся по своим этажам по лестнице, кто-то из сотрудников поднимается вверх, кто-то спускается вниз. На подъём на один этаж сотрудник тратит A секунд, спуск на один этаж занимает B секунд. Лифт тратит C секунд на подъём на один этаж.

Определите, на какой этаж необходимо отвезти сотрудников на лифте, чтобы они попали на свои этажи как можно быстрее (с учётом того времени, которое они потратят на подъём на лифте).

Первая строка входных данных содержит число N – количество этажей в бизнес-центре. Следующие три строки содержат числа A , B , C – время, необходимое сотруднику на подъём на один этаж, на спуск на один этаж, и время, необходимое лифту на подъём на один этаж. Все числа целые положительные, не превосходящие 2×10^9 , при этом $A > B$, $A > C$.

Программа должна вывести единственное целое число – номер этажа, на который должен ехать лифт, чтобы все сотрудники попали на свой этаж как можно быстрее.

Пример входных и выходных данных

Ввод	Вывод	Примечание
6 20 10 5	4	В здании 6 этажей. Сотрудник поднимается на один этаж за 20 секунд, спускается за 10 секунд. Лифт поднимается на один этаж за 5 секунд. Чтобы всем сотрудникам быстрее добраться до своих мест, лифт едет на 4-й этаж за 20 секунд. Сотрудник, который работает на 6-м этаже, выходит из лифта и поднимается за 40 секунд, всего его путь занимает 60 секунд. Сотрудник, работающий на 1-м этаже, спускается на 3 этажа, это занимает $20 + 30 = 50$ секунд. При другом выборе этажа потребуется большее время.

Система оценивания

Решение, правильно работающее только для случаев, когда N не превосходит 100, будет оцениваться в 6 баллов.

Задача 7. Турнир

В турнире участвуют N команд. Турнир проводится по олимпийской системе (команды играют на вылет, проигравшие команды выбывают из турнира, выигравшие проходят в следующий тур, ничьих не бывает). Число команд в этой задаче будет степенью двойки: $N = 2^k$.

Все команды пронумерованы числами от 1 до N . В первом туре играют команды с номерами 1 и 2, 3 и 4, 5 и 6 и т. д., всего играется $N/2$ матчей. По результатам этих матчей команды выходят во второй тур. Во втором туре играют победители первой и второй игры первого тура, победители третьей и четвёртой игры первого тура и т. д. Они выходят в третий тур. В третьем туре играют вместе победители первой и второй игры второго тура, победители третьей и четвёртой игры второго тура и т. д.

Вам даны результаты всех матчей. Определите номер команды, которая стала победителем турнира.

В первой строке входных данных записано число N – количество команд, участвовавших в турнире. Оно является степенью двойки и может принимать значения от $2^0 = 1$ до $2^{16} = 65536$. Следующая $N - 1$ строка содержат результаты всех сыгранных

матчей. Первые $N/2$ строк из них являются результатами матчей первого тура, затем идёт $N/4$ строк с результатами второго тура, $N/8$ строк с результатами третьего тура и т. д.

Результат каждого матча является одним из двух возможных чисел: 1 или 2. Число 1 означает, что в матче выиграла первая команда (номер которой меньше), число 2 означает, что в матче выиграла вторая команда (номер которой больше).

Программа должна вывести одно число – номер победившей в турнире команды.

Пример входных и выходных данных

Ввод	Вывод
8 1 2 2 1 2 1 1	4

Система оценивания

Решение, правильно работающее только для случаев, когда $N \leq 8$, будет оцениваться в 4 балла.

Пояснение к примеру

Далее нарисована схема турнира для примера из условия. В турнире участвовало 8 команд. Результаты матчей: 1, 2, 2, 1, 2, 1, 1.

В первом туре играли команды 1 и 2, 3 и 4, 5 и 6, 7 и 8. Результаты матчей первого тура: 1, 2, 2, 1, во второй тур вышли команды 1, 4, 6, 7.

Во втором туре играли команды 1 и 4, 6 и 7. Результаты матчей второго тура: 2, 1. В третий тур вышли команды 4 и 6.

В последнем, третьем, туре играют команды 4 и 6, результат матча: 1, поэтому победителем турнира является команда 4.

