

# Общая информация по задачам олимпиады

## Правила олимпиады

Олимпиада длится три часа. За это время вам будет предложено решить 7 задач по программированию. Решение задачи — это исходный код программы, которая без ошибок выполняет поставленную задачу.

Каждое решение будет проверено автоматически на некотором заранее заготовленном наборе тестов. Если ваша программа проходит все тесты, она считается решённой верно. Если ваша программа не проходит хотя бы один тест, она считается не решённой.

Штраф за задачу — это время сдачи задачи от начала соревнования в минутах плюс 20 штрафных минут за каждую неуспешную попытку сдачи. В первую очередь, старайтесь решить как можно больше задач, а во вторую сдавайте быстро и без ошибок, чтоб сделать штрафное время как можно меньше.

Задачи упорядочены по сложности, однако сложность — субъективное мнение жюри. Поэтому **обязательно прочтите все задачи!**

## Ввод и вывод

Помните, что на олимпиаде нет ручной проверки Ваших программ. Поэтому не выводите приветственные фразы (например, «Введите число  $n$ :»).

Соблюдайте формат ввода, описанный в соответствующем разделе условий задач. Данные будут введены с теми ограничениями, которые описаны в условиях. Не надо проверять «правильность» входных данных — об этом уже позаботилось жюри.

Выводить нужно именно то, что просят. Если требуется вывести целое число, вы получите ошибку при попытке вывести число с точкой или запятой. Если требуется вывести одну строку, вы получите ошибку при попытке вывести две строки. И так далее...

## Примеры

Гарантируется, что примеры содержат правильный ответ на задачу. Так, вы можете у себя на компьютере протестировать программу ещё до того, как отправите её на проверку.

Также гарантируется, что тесты из условия — это первые тесты в задаче. Так, если вы получаете ошибку в системе на первом тесте (например, `wrong answer 1`), то вы получаете ошибку именно на первом тесте из условия. В этом случае, попробуйте запустить программу у себя на компьютере.

Вы можете убедиться, что ваше решение проходит тесты из условий в системе: для этого отправьте его на *тестирование*.

## Ограничения

Ограничение по памяти во всех задачах — 64 Мегабайта.

Ограничение по времени во всех задачах — 0.5 секунд.

Ограничения по времени и памяти распространяются на один конкретный тест. Обратите внимание, что время выполнения программы в системе может отличаться от времени выполнения на вашем компьютере (обычно система быстрее).

Вы не ограничены по количеству попыток, но вы можете отправлять решения только один раз в десять секунд.

## Прочее

Не забывайте, что вы можете задать вопрос по задаче в системе.

Дополнительную информацию вы можете получить в «справке» в системе. Обратите внимание, что там находится **информация о вердиктах**: если вы получили вердикт по задаче, смысл которого вы не понимаете (или хотите уточнить), прочтите соответствующий пункт в справке.

## Задача А. ESMS

Уже почти всё готово для введения нового стандарта «Even Shorter Message Service» (сокращённо: ESMS) — системы передачи сообщений в мобильных сетях.

В отличие от SMS, в котором длина сообщения была ограничена 160 символами, в ESMS можно отправлять сообщения длиной не более  $B$  символов.

Вам поручили реализовать важный модуль для пользовательского интерфейса — программу, которая по длине уже написанного пользователем сообщения  $A$  и максимальной допустимой длине сообщения  $B$  вычисляет, сколько ещё символов можно добавить в сообщение, чтобы не превысить ограничение длины.

### Формат входных данных

В первой строке вводится целое число  $A$  — длина уже написанного пользователем сообщения ( $0 \leq A < 160$ ).

Во второй строке вводится целое число  $B$  — максимально допустимая длина сообщения ( $0 < B < 160$ ).

Гарантируется что  $A \leq B$ .

### Формат выходных данных

Выведите единственное целое число — количество символов, которые можно добавить в сообщение, не превысив ограничение длины.

### Примеры

тест	ответ
50 100	50
33 33	0

### Замечание

В первом примере пользователь уже написал 50 символов, а ограничение на длину составляет 100 символов. Значит, можно написать ещё 50 символов, не превысив ограничение длины.

Во втором примере пользователь использовал все 33 доступных ему символа.

## Задача В. Табуретка

Стояла табуретка на четырёх ножках, причём длина каждой ножки была ровно  $k$  см.

Елене Вениаминовне стало скучно, и она отпилила кусок от одной из ножек. После этого её длина стала равна ровно  $t$  см.

Если длина пострадавшей ножки не менее 90% от исходной, то на табуретке безопасно сидеть. Безопасно ли сидеть на табуретке?

### Формат входных данных

В первой строке вводится целое число  $k$  — исходная длина каждой ножки ( $1 \leq k \leq 100$ ).

Во второй строке вводится целое число  $t$  — длина пострадавшей ножки ( $0 \leq t < k$ ).

### Формат выходных данных

В единственной строке выведите единственное слово «SAFE» (без кавычек), если на табуретке сидеть безопасно, и «DANGER» (без кавычек), если нет.

### Примеры

тест	ответ
10 9	SAFE
5 4	DANGER

### Замечание

В первом примере длина пострадавшей ножки составляет 90% от исходной, а значит на табуретке сидеть безопасно.

Во втором примере длина пострадавшей ножки составляет 80% от исходной, так что на табуретке сидеть ни в коем случае нельзя!

## Задача С. Партия в теннис

Даниил пришел посмотреть, как его друзья играют в теннис. Его попросили вести счёт игры, однако Даниил не был знаком с необычными правилами начисления баллов в теннисе, поэтому считал как обычно: кто в раунде победил, тому и одно очко (как в футболе или хоккее).

Однако, в теннисе не всё так просто. Вместо стандартных  $0, 1, 2, 3, \dots$  здесь используются *love, 15, 30, 40, \dots*. Более подробно правила опишем так:

Обычный счёт	В теннисе	Обычный счёт	В теннисе
0 : 0	love all	2 : 0	30 love
0 : 1	love 15	2 : 1	30 15
0 : 2	love 30	2 : 2	30 all
0 : 3	love 40	2 : 3	30 40
1 : 0	15 love	3 : 0	40 love
1 : 1	15 all	3 : 1	40 15
1 : 2	15 30	3 : 2	40 30
1 : 3	15 40		

Если текущий обычный счёт есть в таблице, то ему сопоставляется соответствующий теннисный счёт.

Если текущего счёта в таблице нет, то используется следующее правило:

- Если счёт игрока слева равен счёту игрока справа, то в теннисе говорят «**deuce**», рус. — «равно».
- Если счёт игрока слева больше счёта игрока справа, то в теннисе говорят «**advantage in**», рус. — «больше».
- Если счёт игрока слева меньше счёта правого, то в теннисе говорят «**advantage out**», рус. — «меньше».

Партия в самом разгаре, поэтому Даниил не хочет отвлекаться. Помогите ему перевести обычный счёт в теннисный.

### Формат входных данных

В первой строке вводится число  $a$  — сколько очков у игрока слева, если считать по-обычному ( $1 \leq a \leq 100$ ).

Во второй строке вводится число  $b$  — сколько очков у игрока справа, если считать по-обычному ( $1 \leq b \leq 100$ ).

### Формат выходных данных

Выведите английский теннисный счёт по правилам, описанным в условиях. Не выводите кавычки или знаки препинания.

### Примеры

тест	ответ
1 2	15 30
2 2	30 all
3 3	deuce

### Замечание

Разберём первый пример. Обычный счёт сейчас равен  $1 : 2$ . Такой счёт есть в таблице, поэтому мы можем просто вывести соответствующий теннисный счёт: «15 30». Обратите внимание, что ответы: «advantage out», «15:30» — неправильные.

## Задача D. Нули и буквы

Так уж сложилось, что Витя и Ваня общаются только при помощи заглавных и строчных латинских букв «O».

У Вани сломалась клавиша `shift`, поэтому он решил писать вместо заглавных «O» нули.

Витя ненавидит нули, поэтому, увидев сообщение Вани  $s$ , он очень разозлился, и теперь он просит Вас помочь ему: замените все нули в строке  $s$  на заглавные буквы «O».

### Формат входных данных

В первой строке вводится целое число  $n$  — длина строки  $s$  ( $1 \leq n \leq 100$ ).

Во второй строке вводится сообщение  $s$ , состоящее ровно из  $n$  символов: нулей и строчных латинских «o».

Гарантируется, что в строке нет других символов, в частности знаков препинания, пробелов.

### Формат выходных данных

Выведите исправленное сообщение  $s$ , в котором все нули заменены на заглавные латинские буквы «O».

### Пример

тест	ответ
5 o0o0o	oOoOo

## Задача Е. Счастливая улица

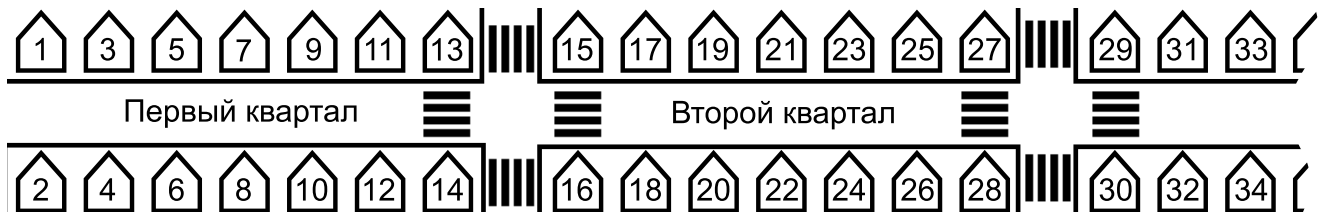
Счастливая улица состоит из  $k$  кварталов. Вы точно знаете, что  $k = 100$ .

Квартал — это 7 домов с одной стороны улицы и 7 — с другой.

На улице есть нумерация домов: все дома на одной стороне нумеруются последовательными нечётными числами, а на другой — чётными.

Любые два соседних квартала соединены перекрёстками. На перекрёстках расположены пешеходные переходы. При помощи одного перехода можно переместиться в соседний квартал на той же стороне улицы или перейти на противоположную сторону.

Для лучшего понимания, посмотрите на рисунок:



Витя живёт в доме с номером  $a$ , но ему надо в дом под номером  $b$ . Он никогда не пойдёт по проезжей части: он либо пойдёт вдоль улицы, либо через переход. Через переходы он ходить ужасно боится, поэтому он просит вас узнать, через какое наименьшее число переходов он должен перейти, чтоб добраться из дома  $a$  в дом  $b$ .

### Формат входных данных

Первая строка содержит целое число  $a$  ( $1 \leq a \leq 1\,400$ ) — дом Вити.

Вторая строка содержит целое число  $b$  ( $1 \leq b \leq 1\,400$ ) — дом, в который Вите надо.

Гарантируется, что  $a \neq b$ .

### Формат выходных данных

Выведите единственное целое число — ответ на задачу.

### Примеры

тест	ответ
7 13	0
11 18	2

### Замечание

В первом примере можно не переходить через дорогу, если идти так:

$$7 \rightarrow 9 \rightarrow 11 \rightarrow 13$$

Во втором примере один из оптимальных путей следующий:

$$11 \rightarrow 13 \xrightarrow{\text{ч/з дорогу}} 15 \xrightarrow{\text{ч/з дорогу}} 16 \rightarrow 18$$

## Задача F. Весёлая компания

Весёлая компания собралась на сплав. По предварительной договорённости каждый из  $n$  человек весёлой компании взял **ровно по одному веслу**.

Оказалось, что из всех взятых вёсел  $l$  подходят только для левой руки,  $r$  — только для правой, а остальные — универсальны.

Весёлая компания решила перераспределить вёсла так, чтобы у каждого человека вёсел либо не было, либо было ровно 2 — по одному в руку (гарантируется, что у каждого человека ровно одна рука левая и ровно одна правая). При этом некоторые вёсла могут остаться без человека: их придётся пустить на дрова.

Людям не нравится, когда принесённые ими вёсла отправляют на дрова. Поэтому компания хочет узнать, какое наименьшее число вёсел придётся пустить на дрова при описанном выше распределении.

### Формат входных данных

В первой строке находится целое число  $n$  — количество людей в компании ( $0 \leq n \leq 10^9$ ).

Во второй строке вводится целое число  $l$  — количество левых вёсел ( $0 \leq l \leq 10^9$ ).

В третьей строке вводится целое число  $r$  — количество правых вёсел ( $0 \leq r \leq 10^9$ ).

Гарантируется, что  $l + r \leq n$ .

### Формат выходных данных

Выведите единственное целое число — наименьшее количество вёсел, которые придётся пустить на дрова.

### Примеры

тест	ответ
11 6 5	1
10 0 8	6

### Замечание

В первом примере дадим 5 людям по левому и по правому веслу. Таким образом, останется только одно неиспользованное левое весло. Увы, его придётся пустить на дрова.

Во втором примере дадим двум людям универсальное весло в левую руку и правое весло в правую руку. После этого останется шесть неиспользованных правых вёсел, которые мы пустим на дрова.

## Задача G. Сложить единицы

Пусть  $f(k) = \underbrace{111\dots 111}_k$ .

Ответьте на  $q$  запросов. Каждый запрос состоит из числа  $k$  и  $n$  чисел:  $a_1, a_2, \dots, a_n$ . Выведите для каждого запроса  $k$ -ю цифру в числе  $f(a_1) + f(a_2) + \dots + f(a_n)$ . Цифры при этом будем считать справа налево (от младшего разряда к старшему), начиная с единицы.

### Формат входных данных

В первой строке вводится целое число  $q$  — количество запросов (в тесте из условия  $q = 2$ , а во всех остальных тестах  $q = 10$ ).

Дальше вводятся  $q$  блоков данных.

В первой строке каждого блока вводится целое число  $k$  — цифра, значение которой требуется узнать ( $1 \leq k \leq 10^{18}$ ).

Во второй строке вводится целое число  $n$  — количество слагаемых ( $1 \leq n \leq 9$ ).

Следующие (и последние)  $n$  строк блока содержат по одному целому числу:  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^{18}$ ).

Гарантируется, что число  $f(a_1) + f(a_2) + \dots + f(a_n)$  содержит как минимум  $k$  цифр.

### Формат выходных данных

Выведите  $q$  целых чисел:  $i$ -е число равно ответу на  $i$ -й запрос.

### Пример

тест	ответ
2	2
1	1
2	
1	
2	
3	
2	
5	
2	

### Замечание

Первый запрос — нужно найти первую цифру числа  $f(1) + f(2) = 1 + 11 = 12$ . У числа 12 две цифры: первая цифра 2, а вторая — 1. Таким образом, ответ на запрос: 2.