

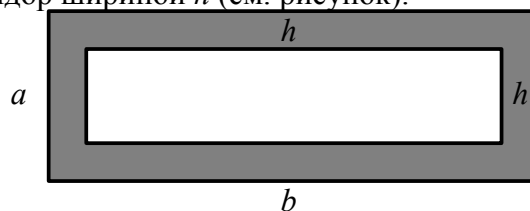
Всероссийская олимпиада школьников по информатике, 2017-18 уч. год
Тренировочный тур для школьного этапа, г. Москва
Задания для 7-8 классов

Каждая задача оценивается в 10 баллов. Итоговый балл выставляется как сумма баллов за 4 задачи с лучшим результатом (то есть для получения максимального балла нужно решить 4 любые задачи). Вы можете сдать не более 50 решений суммарно по всем задачам.

В задачах 1–4 на проверку нужно сдать ответ, имеющий вид некоторой строки текста или последовательности чисел. Оценивается последнее сданное решение, которое удовлетворяет формату записи ответа, приведённому в условии.

Задача 1. Ковровая дорожка

План здания имеет вид прямоугольника со сторонами $a \times b$. Вдоль всех стен здания (внутри здания) проходит коридор шириной h (см. рисунок).



Весь коридор решили покрыть ковровой дорожкой. Определите площадь дорожки. Считайте, что $a > 2h$ и $b > 2h$.

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

Пример правильного (по форме записи) выражения: $a + (b - h) * 2$

Задача 2. Маски имён файлов

Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов. Также в маске может встречаться символ «*», обозначающий любую последовательность любых символов, возможно пустую. Например, если задана маска « $a*po*$ », то такой маске удовлетворяют файлы с именами «**airport**», «**appoint**», «**apologize**», но не удовлетворяют, например, файлы с именами «**approve**» или «**weapon**».

Вам нужно придумать маску имени файла, которой удовлетворяли бы следующие имена файлов:

winter_skates

и

autumn_leafs

но при этом **НЕ** удовлетворяли бы файлы с именами

spring_weather

и

summer_holidays

Кроме того, вам нужно придумать как можно более короткую маску (содержащую как можно меньше символов). Чем короче будет придуманная вами маска, тем больше баллов вы получите. Ответом на эту задачу является строка, которая может содержать строчные буквы английского алфавита, символ «_» (подчёркивание) и символ «*» (звёздочка).

Задача 3. Монетки

Центральный банк для удобства денежного обращения хочет выпустить новые монеты, номинал каждой монеты должен измеряться целым числом рублей.

Центральный банк считает систему номиналов монет удобной, если любую сумму от 1 до 10 рублей можно заплатить при помощи одной монеты или двух монет (они могут быть как одинакового номинала, так и различного номинала). Также Центральный банк стремится, чтобы количество выпущенных номиналов монет было как можно меньше.

Помогите Центральному банку решить эту задачу.

Ответом к этой задаче является строка из целых чисел, записанных через пробел, в которой перечислены номиналы монет, выпущенных Центральным банком. Числа должны быть различными, от 1 до 10.

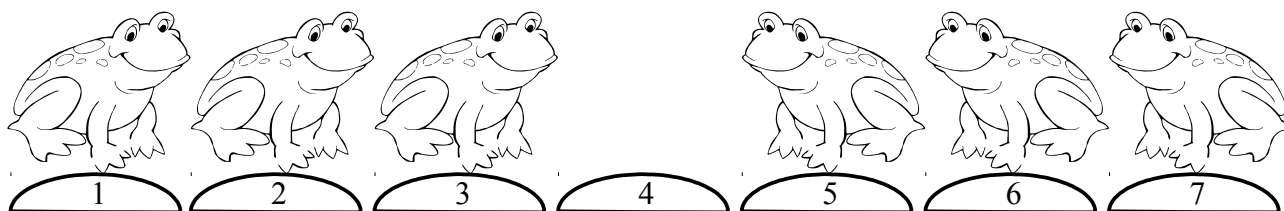
Чем меньше различных чисел (номиналов монет) будет в вашем ответе, тем больше баллов вы получите. Но, разумеется, только в том случае, если предложенный вами набор номиналов удовлетворяет требованию Центрального банка.

Задача 4. Лягушки

На болоте – семь кочек в линию, пронумерованных слева направо числами от 1 до 7. На трёх левых кочках (их номера 1–3) сидят три зелёные лягушки и смотрят направо, на трёх правых кочках (их номера 5–7) сидят три коричневые лягушки и смотрят налево. Лягушка может прыгнуть с той кочки, на которой она сидит, на свободную кочку, но только если эта кочка соседняя с ней либо если при этом нужно перепрыгнуть через одну другую кочку. Перепрыгивать через две кочки и более лягушка не может, также лягушки прыгают всегда только вперёд (по направлению своего взгляда) и не могут разворачиваться. Например, в самом начале игры прыгать могут только лягушки, сидящие на кочках номер 2, 3, 5, 6.

Вам необходимо поменять лягушек местами: зелёные лягушки должны оказаться на кочках 5, 6, 7, а коричневые – на кочках 1, 2, 3.

Вы можете попробовать поиграть в эту игру на странице gg.gg/frogs.



Ответ на эту задачу нужно записать в виде последовательности чисел от 1 до 7, записанных в строку через пробел. Каждое число в последовательности является номером кочки, на которой сидит лягушка, совершающая очередной прыжок. Лягушка прыгает на свободную кочку, если это возможно по правилам игры. Если это невозможно (например, если пустая кочка не является соседней, или не находится через одну в направлении движения лягушки, или если на указанной кочке не сидит лягушка), то команда перемещения считается некорректной и алгоритм прерывается.

Решение получит максимальный балл, если зелёные лягушки будут сидеть на кочках 5, 6, 7, а коричневые — на кочках 1, 2, 3. Если данная цель не будет достигнута, то решение получит тем больше баллов, чем больше лягушек удастся поменять местами.

Решением задач 5–7 является программа, написанная на одном из языков программирования. Задачи 5–7 необязательно решать для получения полного балла.

Ограничение по времени работы программы в задачах 5–7: 1 секунда.

Задания оцениваются, только если они выдают правильный ответ на всех примерах входных и выходных данных, приведённых в условии задачи. Программа не должна выводить никаких иных сообщений, кроме того, что требуется найти в задаче. Во всех задачах целые числа во входных и выходных данных записываются только цифрами (то есть недопустимо использование записи 1000000.0 или 1e6 вместо числа 1000000). Каждое число во входных данных записано в отдельной строке.

Каждое сданное решение задач 5–7 проверяется сразу же после его отправки в тестирующую систему. По каждой задаче 5–7 засчитывается решение, набравшее наибольшее число баллов.

Задача 5. Считалка

Для выбора водящего в детской игре N человек становятся в круг, после чего произносится считалка. На первом слове считалки указывается на первого человека в кругу, на втором слове – на второго человека и т. д. После N -го человека снова идёт первый человек (все люди в кругу пронумерованы числами от 1 до N , круг заикливаются, после человека с номером N идёт человек с номером 1).

Всего в считалке M слов. Определите, на какого человека придётся последнее слово считалки.

Программа получает на вход два целых положительных числа. Первое число N – количество людей в кругу. Второе число M – количество слов в считалке. Оба числа не превосходят 10^9 .

Программа должна вывести одно целое число от 1 до N – номер человека в кругу на которого придётся последнее слово считалки.

Пример входных и выходных данных

Ввод	Вывод
10 25	5

Система оценивания

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

Примеры оформления решения задачи

Ниже даны примеры ввода и вывода данных к этой задаче на нескольких языках программирования. Выберите один из языков программирования, допишите соответствующую программу и отправьте её на проверку с использованием одного из допустимых компиляторов.

Алгоритмический язык Интерпретатор Кумир	Язык программирования Pascal Компиляторы Free Pascal, Borland Delphi, Pascal ABC.NET
алг Задач1 нач цел n, m, result ввод n ввод m ... result := ...	var n, m, result: longint; begin readln(n); readln(m); ... result := ... writeln(result);

<pre> вывод result кОН </pre>	<pre> end. </pre>
<p>Язык программирования Basic Компилятор Free Basic (аналог qbasic)</p> <pre> DIM n, m, result AS INTEGER INPUT n INPUT m ... result = ... print result </pre>	<p>Язык программирования Basic Компилятор Mono Visual Basic</p> <pre> Module ProgramA Sub Main() DIM n, m, result AS INTEGER n = Cint(Console.ReadLine()) m = Cint(Console.ReadLine()) ... result = ... Console.WriteLine(CStr(result)) End Sub End Module </pre>
<p>Язык программирования C Компилятор GNU C</p> <pre> #include<stdio.h> int main() { int n, m, result; scanf("%d%d", &n, &m); ... result = ... printf("%d", result); return 0; } </pre>	<p>Язык программирования C++ Компилятор GNU C++</p> <pre> #include<iostream> using namespace std; int main() { int n, m, result; cin >> n >> m; ... result = ... cout << result; return 0; } </pre>
<p>Язык программирования Python Пример для версий 2 и 3 языка Python</p> <pre> n = int(input()) m = int(input()) ... result = ... print(result) </pre>	<p>Язык программирования C# Компилятор Mono C#</p> <pre> using System; using System.IO; class Program { static void Main() { int n, m, result; n = int.Parse(Console.ReadLine()); m = int.Parse(Console.ReadLine()); ... result = ... Console.WriteLine("{0}", result); } } </pre>
<p>Язык программирования PHP Работает в режиме CLI (без web-сервера)</p> <pre> <?php \$n = fgets(STDIN); \$m = fgets(STDIN); ... \$result = ... print \$result; ?> </pre>	<p>Язык программирования Java</p> <pre> import java.io.*; public class Main { public static void main(String[] args) throws Exception { DataInputStream in = new DataInputStream(System.in); int n, m, result; n = Integer.parseInt(in.readLine()); m = Integer.parseInt(in.readLine()); ... result = ... System.out.println(result); } } </pre>

Задача 6. Конфеты

На столе стоят три вазы с конфетами. В левой вазе лежат A конфет, в средней вазе лежат B конфет, в правой вазе лежат C конфет. Лена съедает одну конфету из левой вазы, затем – одну конфету из средней вазы, затем из правой, средней, левой, средней, правой, средней и т. д. (слева направо, затем налево, опять направо и т.д.)

Если Лена хочет взять конфету из какой-то вазы, а конфет там нет, она расстраивается и идёт спать. Определите, сколько конфет съест Лена.

Программа получает на вход три целых неотрицательных числа A, B, C – количество конфет в левой, средней, правой вазе. Сумма трёх данных чисел не превосходит 2×10^9 .

Пример входных и выходных данных

Ввод	Вывод	Примечание
3 3 3	7	Лена съест конфеты из левой, средней, правой, средней, левой, средней, правой вазы. После этого она захочет съесть конфету из средней вазы, но в ней уже не осталось конфет.

Система оценивания

Решение, правильно работающее только для случаев, когда входные числа не превосходят 10, будет оцениваться в 4 балла.

Решение, правильно работающее только для случаев, когда входные числа не превосходят 10 000, будет оцениваться в 7 баллов.

Задача 7. Расписание кружка

Володе очень понравились задачи олимпиады по информатике, поэтому он решил ходить на занятия кружка по программированию. Придя на первое занятие кружка, он узнал, что занятия будут проходить еженедельно в один и тот же день недели. Помогите Володе составить календарь занятий до конца года – определите даты всех занятий, начиная с первого занятия и до конца года.

Программа получает на вход два числа, записанных в разных строках: номер месяца и номер дня месяца, когда проходит первое занятие. Номер месяца может быть одним из четырёх возможных чисел – 9, 10, 11, 12. Номер дня месяца – число от 1 до 30 для сентября и ноября (месяцы с номерами 9 и 11) или от 1 до 31 для октября и декабря (месяцы с номерами 10 и 12).

Программа должна вывести даты всех занятий кружка до конца года в хронологическом порядке, по одной дате в строке, сначала месяц, затем день месяца, через пробел. Занятия проходят еженедельно, в тот же день недели, что и первое занятие. Формат вывода дат такой же, как в условии. Считайте, что каникулы отсутствуют, а последнее занятие может происходить в любой день декабря, в том числе и 31 числа.

Пример входных и выходных данных

Ввод	Вывод
11	11 20
20	11 27
	12 4
	12 11
	12 18
	12 25