



**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ**
**ОЛИМПИАДА ШКОЛЬНИКОВ САНКТ-ПЕТЕРБУРГСКОГО
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА**
Общеобразовательный предмет/ комплекс предметов: Информатика
2011-2012 учебный год

Отборочный этап

Задача 1

Дано два целых положительных числа: **a** и **b**, $1 \leq a, b \leq 10000$.
Напишите программу, которая находит цифру, на которую оканчивается число a^b .

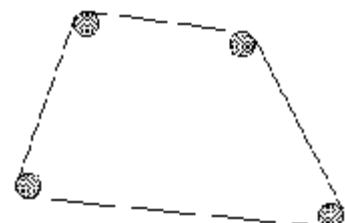
<p>Формат входных данных (input.txt): В единственной строке входного файла записаны числа a и b, разделенные пробелом.</p> <p>Формат выходных данных (output.txt): В выходной файл следует записать одну цифру – ту, на которую оканчивается число a^b.</p>	<p><i>Пример:</i> input.txt: 2 2 output.txt: 4</p>
--	--

Задача 2

Даны **N** различных символов, $1 < N < 9$, каждый из которых представляет собой букву латинского алфавита. Напишите программу, формирующую все возможные «слова» (комбинации букв) длиной **N** букв, получаемые перестановкой заданных символов.

<p>Формат входных данных (input.txt): Первая строка входного файла содержит число N. Вторая строка входного файла содержит исходные символы, разделенные пробелами.</p> <p>Формат выходных данных (output.txt): Каждая строка выходного файла содержит комбинацию букв, составляющую очередное «слово» длиной N.</p>	<p><i>Пример:</i> input.txt: 3 a b c output.txt: abc acb bac bca cab cba</p>
--	--

Задача 3



В плоскую поверхность вбили N гвоздей так, что снаружи остались только шляпки. Все гвозди вбиты в вершины выпуклого многоугольника. После этого вокруг шляпок всех гвоздей была натянута нить (см. рисунок).

Задача – определить длину этой нити, если известны координаты центра каждого гвоздя, а также радиус шляпок (одинаковый для всех гвоздей).

<p>Формат входных данных (input.txt): Первая строка файла содержит два числа – количество гвоздей N, $1 < N < 100$, и вещественное число R – радиус шляпок гвоздей. Все шляпки имеют одинаковый радиус. Далее во входном файле располагаются еще N строк, в каждой из которых записана через пробел пара вещественных координат центра очередного гвоздя; координаты не превосходят по абсолютной величине числа 100. Описания гвоздей приводятся в порядке обхода вершин многоугольника по или против часовой стрелки, начиная с произвольного гвоздя. Шляпки разных гвоздей не соприкасаются.</p> <p>Формат выходных данных (output.txt): Выходной файл должен в своей единственной строке содержать вещественное число, округленное до двух знаков после запятой – длину ниточки, натянутой вокруг всех гвоздей.</p>	<p><i>Пример:</i> input.txt: 4 1 0.0 0.0 2.0 0.0 2.0 2.0 0.0 2.0 output.txt: 14.28</p>
--	--

Задача 4

Написать программу, которая будет находить, на сколько квадратов, стороны которых выражены натуральными числами, можно разрезать данный прямоугольник, если от него каждый раз отрезается квадрат максимальной площади.

<p>Формат входных данных (input.txt): Входной файл состоит из одной строки, содержащей два натуральных числа – стороны прямоугольника, разделенные пробелом.</p> <p>Формат выходных данных (output.txt): Выходной файл содержит одно число – количество квадратов.</p>	<p><i>Пример:</i> input.txt: 6 4 output.txt: 3</p>
--	--

Задача 5

В одной далекой стране зарплату жителям выдали банкнотами достоинством **3** и **5** денежных единицы (*д.е.*). Известно, что для совершения покупки на сумму $n > 8$ *д.е.* существует комбинация банкнот **3** *д.е.* и **5** *д.е.*, такая, чтобы получилась необходимая сумма **n**.

Требуется подсчитать сколько банкнот по **3** *д.е.* и по **5** *д.е.* необходимо для совершения покупки на сумму n *д.е.*

<p><u>Формат входных данных (input.txt):</u> Входной файл состоит из одной строки, содержащей требуемую сумму – целое число n.</p> <p><u>Формат выходных данных (output.txt):</u> Выходной файл содержит два целых числа – количество банкнот 3 <i>д.е.</i> и количество банкнот 5 <i>д.е.</i>, разделенные пробелом.</p>	<p><i>Пример:</i> input.txt: 31 output.txt: 7 2</p>
--	---

Задача 6

Даны **N** различных цифр, $1 < N < 9$, ни одна из цифр не равна нулю. Напишите программу, формирующую все возможные **N**-значные числа, получаемые перестановкой заданных цифр.

<p><u>Формат входных данных (input.txt):</u> Первая строка входного файла содержит число N. Вторая строка входного файла содержит исходные цифры, разделенные пробелами.</p> <p><u>Формат выходных данных (output.txt):</u> Каждая строка выходного файла содержит комбинацию цифр, составляющую очередное N-значное число.</p>	<p><i>Пример:</i> input.txt: 3 1 2 3 output.txt: 123 132 213 231 312 321</p>
---	--



САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ
ОЛИМПИАДА ШКОЛЬНИКОВ САНКТ-ПЕТЕРБУРГСКОГО
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

Общеобразовательный предмет/ комплекс предметов: Информатика
2011-2012 учебный год

Заключительный этап

Задача №1. КВАДРАТЫ

В одном квадратном государстве жили квадратные люди. И все остальное в этом государстве было тоже квадратное. Так, Квадратная Дума приняла Квадратный Закон о земле. Согласно этому закону, любой житель государства имел право приобрести землю. Земля продавалась квадратными участками. Длина стороны каждого участка выражалась натуральным числом метров. Приобретая участок земли со стороной A метров, покупатель платил A^2 квадриков (местная валюта) и получал одно квадратное свидетельство о праве собственности на этот участок. Один житель этого государства решил вложить все свои N квадриков без остатка в покупку земли. Это, безусловно, можно было сделать, приобретя участки размером 1×1 метр. Но этот житель потребовал от агентства недвижимости минимизации количества покупаемых участков. "Так мне будет легче общаться с Квадратной Налоговой Инспекцией", - сказал он. Сделка состоялась.

Задание: Напишите программу, которая находит количество свидетельств (наименьшее из возможных), полученных жителем.

Формат входных данных Единственная строка входного файла содержит целое натуральное число $N \leq 60000$ - число квадриков, которое было у жителя.	Пример: <table border="1"><tr><td>Ввод</td></tr><tr><td>344</td></tr><tr><td>Вывод</td></tr><tr><td>3</td></tr></table>	Ввод	344	Вывод	3
Ввод					
344					
Вывод					
3					
Формат выходных данных Единственная строка выходного файла содержит число свидетельств, полученных в результате сделки.					

Задача №2. ТОЧКИ В МНОГОУГОЛЬНИКЕ

Многоугольник (не обязательно выпуклый) на плоскости задан координатами своих вершин.

Задание: Необходимо написать программу для подсчета количества точек с целочисленными координатами, лежащих внутри многогранника (но не на его границе).

Формат входных данных В первой строке содержится N ($3 \leq N \leq 1000$) - число вершин многоугольника. В последующих N строках идут координаты (X_i, Y_i) вершин многоугольника в порядке обхода по часовой стрелке. X_i и Y_i - целые числа, по модулю не превосходящие 1000000.	Примеры: <table border="1"><tr><td>Ввод</td><td>Ввод</td></tr><tr><td>4</td><td>3</td></tr><tr><td>-1 -1</td><td>0 0</td></tr><tr><td>-1 1</td><td>0 2</td></tr><tr><td>1 1</td><td>2 0</td></tr><tr><td>1 -1</td><td>Вывод</td></tr><tr><td>Вывод</td><td>0</td></tr><tr><td>1</td><td></td></tr></table>	Ввод	Ввод	4	3	-1 -1	0 0	-1 1	0 2	1 1	2 0	1 -1	Вывод	Вывод	0	1	
Ввод	Ввод																
4	3																
-1 -1	0 0																
-1 1	0 2																
1 1	2 0																
1 -1	Вывод																
Вывод	0																
1																	
Формат выходных данных В выходной файл вывести одно число - искомое число точек																	

Задача №3. ХОД КОНЕМ

На шахматной доске (8x8) стоят конь и пешка. Конь располагается на поле А, пешка - на поле В. Найти минимальное количество ходов, за которые конь доберется до пешки, если последняя стоит на месте. За один ход конь выполняет стандартное шахматное перемещение (буквой "Г").

Задание: Необходимо написать программу для подсчета наименьшего количества перемещений (ходов) для перехода коня с поля А на поле В.

<p>Формат входных данных Первая строка содержит координаты поля А - две цифры, разделенные пробелом (первая цифра - номер строки, вторая - номер столбца). Значения координат лежат в диапазоне от 1 до 8. Координаты отсчитываются от левого нижнего угла шахматной доски. Вторая строка содержит координаты поля В (аналогичные по структуре построения).</p> <p>Формат выходных данных В выходной файл записывается число n - найденное наименьшее число ходов.</p>	<p>Пример:</p> <table border="1"><tr><td>Ввод</td></tr><tr><td>3 4</td></tr><tr><td>6 4</td></tr><tr><td>Вывод</td></tr><tr><td>3</td></tr></table>	Ввод	3 4	6 4	Вывод	3
Ввод						
3 4						
6 4						
Вывод						
3						

Задача №4. МАРСИАНСКИЕ УРАВНЕНИЯ

Во время недавних раскопок на Марсе были обнаружены листы бумаги с таинственными символами на них. После долгих исследований учёные пришли к выводу, что надписи на них на самом деле могли быть обычными числовыми равенствами. Если бы этот вывод оказался верным, это доказало бы не только то, что на Марсе много лет назад были разумные существа, но и то, что они уже умели считать.

Ученые смогли понять, что в этом случае означают найденные символы, и перевели эти равенства на обычный язык - язык цифр, скобок, знаков арифметических действий и равенства. Кроме того, из других источников было получено веское доказательство того, что марсиане знали только три операции - сложение, умножение и вычитание (марсиане никогда не использовали "унарный минус": вместо "-5" они писали "0-5"). Также ученые доказали, что марсиане не наделяли операции разным приоритетом, а просто вычисляли выражения (если в них не было скобок) слева направо: например, $3+3*5$ у них равнялось 30, а не 18.

К сожалению, символы арифметических действий марсиане почему-то наносили специальными чернилами, которые, как оказалось, были не очень стойкими, и поэтому в найденных листках между числами вместо знаков действий были пробелы. Если вся вышеизложенная теория верна, то вместо этих пробелов можно поставить знаки сложения, вычитания и умножения так, чтобы равенства стали верными.

Например, если был найден лист бумаги с надписью "18=7 (5 3) 2", то возможна такая расстановка знаков: "18=7+(5-3)*2" (помните про то, в каком порядке марсиане вычисляют выражения!). В то же время, если попался лист с надписью "5=3 3", то марсиане явно не имели в виду числового равенства, когда писали это.

Задание: Необходимо написать программу, находящую требуемую расстановку знаков или сообщающую, что таковой не существует.

<p>Формат входных данных Единственная строка входного файла состоит из натурального (целого положительного) числа, не превосходящего 230, знака равенства, и последовательности натуральных чисел (не более десяти), произведение которых также не превосходит 230. Некоторые группы чисел (одно или более) могут быть окружены скобками. Длина входной строки не будет превосходить 80 символов, и других ограничений на количество и вложенность скобок нет. Между двумя соседними числами, не разделенными скобками, всегда будет хотя бы один пробел, во всех остальных местах может быть любое (в том числе и 0) число пробелов (естественно, внутри числа пробелов нет).</p> <p>Формат выходных данных В выходной файл необходимо вывести одну строку, содержащую полученное равенство (т.е., исходное равенство со вставленными знаками арифметических действий). В случае если требуемая расстановка знаков невозможна, вывести строку, состоящую из единственного числа "-1". Выходная строка не должна содержать пробелов.</p>	<p>Пример №1:</p> <table border="1"><tr><td>Ввод</td></tr><tr><td>18=7 (5 3) 2</td></tr><tr><td>Вывод</td></tr><tr><td>18=7+(5-3)*2</td></tr></table> <p>Пример №2:</p> <table border="1"><tr><td>Ввод</td></tr><tr><td>18=7 (5 3) 2</td></tr><tr><td>Вывод</td></tr><tr><td>18=7+(5-3)*2</td></tr></table>	Ввод	18=7 (5 3) 2	Вывод	18=7+(5-3)*2	Ввод	18=7 (5 3) 2	Вывод	18=7+(5-3)*2
Ввод									
18=7 (5 3) 2									
Вывод									
18=7+(5-3)*2									
Ввод									
18=7 (5 3) 2									
Вывод									
18=7+(5-3)*2									

Задача №5. ДРЕМУЧИЙ ЛЕС

В лесу находится наблюдатель. Будем говорить, что для наблюдателя лес является дремучим, если из своего текущего положения наблюдатель видит только деревья. Вам дана карта леса и координаты точки, в которой находится наблюдатель.

На карте леса все деревья изображаются кругами. При этом в лесу бывают сросшиеся деревья (изображения таких деревьев на карте пересекаются), также одно дерево может находиться внутри другого. Точка, в которой стоит наблюдатель, не лежит внутри или на границе ни одного из деревьев.

Задание: Требуется написать программу, которая определяла, кажется ли лес дремучим данному наблюдателю.

<p>Формат входных данных В первой строке содержится целое число N - количество деревьев ($1 \leq N \leq 50000$). Вторая строка содержит два числа, задающих координаты наблюдателя. Далее следует N строк, содержащих тройки чисел, задающих деревья. Первые два числа задают координаты центра дерева, а третье – радиус дерева. Все координаты и радиус задаются точно, и выражаются вещественными числами не более чем с 2 знаками после десятичной точки, по модулю не превосходящими 100000.</p> <p>Формат выходных данных В единственной строке выходного файла должно содержаться сообщение <i>YES</i>, если лес является дремучим, или <i>NO</i> в противном случае.</p>	<p>Примеры:</p> <table border="1" style="display: inline-table; margin-right: 20px;"><tr><th>Ввод</th></tr><tr><td>2</td></tr><tr><td>10 10</td></tr><tr><td>0 0 1</td></tr><tr><td>0.5 0 2</td></tr><tr><th>Вывод</th></tr><tr><td>NO</td></tr></table> <table border="1" style="display: inline-table;"><tr><th>Ввод</th></tr><tr><td>4</td></tr><tr><td>0 0</td></tr><tr><td>2 2 2</td></tr><tr><td>-2 2 2</td></tr><tr><td>-2 -2 2</td></tr><tr><td>2 -2 2</td></tr><tr><th>Вывод</th></tr><tr><td>YES</td></tr></table>	Ввод	2	10 10	0 0 1	0.5 0 2	Вывод	NO	Ввод	4	0 0	2 2 2	-2 2 2	-2 -2 2	2 -2 2	Вывод	YES
Ввод																	
2																	
10 10																	
0 0 1																	
0.5 0 2																	
Вывод																	
NO																	
Ввод																	
4																	
0 0																	
2 2 2																	
-2 2 2																	
-2 -2 2																	
2 -2 2																	
Вывод																	
YES																	

Задача №6. КОЛЛЕКЦИЯ

Вася коллекционирует спичечные этикетки. Для этого у него есть N альбомов вместимостью K_1, K_2, \dots, K_N этикеток. Вася хочет, чтобы в случае утери одного любого альбома каждая этикетка осталась у него хотя бы в одном экземпляре. Для этого он покупает каждую этикетку в двух экземплярах, и наклеивает их в два разных альбома.

Задание: Требуется написать программу, которая находила максимальное количество различных этикеток которое может оказаться в описанной коллекции.

<p>Формат входных данных Первая строка содержит число N - количество альбомов (N - натуральное число из диапазона от 2 до 1000). Вторая строка содержит N чисел: K_1, K_2, \dots, K_N, разделенных пробелом, задающих вместимости альбомов. Вместимость каждого альбома задается натуральным числом, суммарная вместимость всех альбомов не превышает 100000 этикеток.</p> <p>Формат выходных данных В выходной файл выведите число E - максимальное количество различных этикеток, которое может собрать Вася с соблюдением выдвинутого условия.</p>	<p>Пример:</p> <table border="1" style="margin-left: auto; margin-right: auto;"><tr><th>Ввод</th></tr><tr><td>4</td></tr><tr><td>1 2 1 1</td></tr><tr><th>Вывод</th></tr><tr><td>2</td></tr></table>	Ввод	4	1 2 1 1	Вывод	2
Ввод						
4						
1 2 1 1						
Вывод						
2						