

Фестиваль «Юные интеллектуалы Среднего Урала»
Муниципальный этап Всероссийской олимпиады по информатике
2010–2011 учебный год
11 класс

Время выполнения задач — 4 часа
Ограничение по времени — 2 секунды на тест
Ограничение по памяти — 64 мегабайта

Уважаемый участник!

В настоящее время решения на олимпиадах по информатике проверяются автоматически. Ваша задача — написать программу, которая по заданным входным данным вычисляет и выводит выходные данные. Когда вы сдаёте решение на проверку, проверяющая программа «подсовывает» вашему решению тестовые наборы входных данных, запускает вашу программу и затем анализирует выданный ею результат: если в задаче возможно несколько правильных ответов, то программа может выводить любой из них, если в условии не указано иное.

Ваше решение должно читать входные данные из файла **input.txt** в описанном формате, решать задачу, и выводить результат в файл **output.txt**. Всякая строка в файле **input.txt** завершается переводом строки. Программа должна всегда завершаться с кодом 0 (иначе тестирующая программа считает, что в ходе работы произошла ошибка) — то есть, командой **halt(0)**; или просто достижением конца текста, если Вы пишете на Паскале, или **return 0**; для программ на C/C++.

Считывать информацию из текстового файла так же просто, как и с клавиатуры — в Бейсике, Паскале и C/C++ для ввода с клавиатуры и из файла используются или одни и те же операторы или же операторы со сходной структурой вызова. То же касается и вывода в файл. Возникают лишь небольшие отличия, связанные с файловыми переменными (см. памятку по работе с файлами).

Ваша программа не должна ничего выводить на экран, а также ждать какого-либо ввода пользователя. Распространённой ошибкой является ситуация, когда после окончания работы программа ждёт нажатия какой-либо клавиши. Этого быть не должно.

Те, кто программирует на Паскале в среде Borland Pascal, не должны использовать модуль CRT: программа не должна содержать команды **uses crt**;

Программа должна выдавать в выходной файл ту и только ту информацию, которая описана в формате вывода. Более того, вывод программы должен в точности удовлетворять формату, описанному в условии задачи (заглавные/строчные буквы, наличие/отсутствие пробелов, переводы строк). В противном случае, программа считается нерабочей и оценивается в 0 баллов.

Все задачи считаются равноценными и имеют одинаковую максимальную оценку в 100 баллов. Все тесты в рамках каждой задачи также равноценны.

11.1. «Двойной факториал». Помогите Пете Торопыжкину написать программу, которая по введённому целому числу n вычислит $n!!$. Напомним, что

$$n!! = \begin{cases} 1 \cdot 3 \cdot \dots \cdot n, & \text{если } n \text{ нечётно,} \\ 2 \cdot 4 \cdot \dots \cdot n, & \text{если } n \text{ чётно.} \end{cases}$$

Формат ввода: В единственной строке записано целое число n ($1 \leq n \leq 20$).

Формат вывода: Выведите единственное целое число — результат вычисления.

Пример 1

input.txt:	output.txt:
3	3

Пример 2

input.txt:	output.txt:
4	8

11.2. «На уроке английского-2». В тетрадке по английскому языку Петя Торопыжкин написал длинное слово, состоящее из заглавных букв латинского алфавита. Его заинтересовали *подслова* этого слова (то есть, подстроки данной строки). Двухбуквенные подслова он изучил быстро, а с трёхбуквенными вышла заминка. Помогите Пете: напишите программу, которая по заданному слову выдаёт его подслово из трёх букв, максимальное в лексикографическом порядке. Напомним, что *лексикографическим порядком* называется тот порядок, в котором упорядочены слова в словаре: сначала сравниваются первые буквы, в случае их равенства — вторые, а в случае равенства вторых — третьи.

Формат ввода: Единственная строка содержит слово, которое написал Петя. Это слово состоит только из заглавных латинских букв и имеет длину не менее 3 и не более 200 символов. Слово завершается переводом строки.

Формат вывода: Выведите в единственной строке требуемое трёхбуквенное подслово.

Пример

input.txt:	output.txt:
ABCA	BCA

11.3. «Ночной Дозор». Возможно, вам известна игра «Ночной Дозор». Вот один из вариантов правил. Выделяется n локаций на местности. Команда приезжает в первую локацию, где получает первую загадку. Отгадывая её, команда получает указание, в какую локацию ехать дальше, переезжает туда, снова отгадывает загадку, переезжает на новое место и так далее. Игра заканчивается, когда команда достигает одной из локаций, отмеченных организаторами как финальные.

Организаторы очередной игры составили план перемещения по локациям в виде списка чисел k_i . Число k_i указывает, что из i -й локации нужно следовать в локацию с номером k_i . Если i -я локация финальная, то $k_i = -1$.

Сейчас они хотят проверить, завершит ли за разумное время свои поездки команда, приехавшая в начале игры на первую локацию.

Формат ввода: В первой строке записано целое число n — количество локаций ($1 \leq n \leq 1000$). В i -й из следующих n строк записано число k_i ($1 \leq k_i \leq n$ или $k_i = -1$).

Формат вывода: Выведите единственное целое число — количество переездов, которое придётся совершить команде, приехавшей в начале игры на первую локацию, для того, чтобы достичь какой-либо финальной локации, либо -1, если план переездов составлен так, что команда, следуя ему, никогда не достигнет никакой финальной локации.

Пример 1

input.txt:	output.txt:
3	2
2	
3	
-1	

Пример 2

input.txt:	output.txt:
3	-1
2	
1	
-1	

11.4. «Магнитное число». Родители Пети Торопыжкина купили магниты на холодильник, имеющие форму цифр: a_0 нулей, a_1 единиц, \dots , a_9 девяток. Когда мама сказала, что пора идти спать, Петя ответил, что хочет сначала составить из магнитов минимально возможное целое положительное число, кратное 15, так, чтобы все магниты были задействованы. Напишите программу, которая по набору магнитов у Пети найдёт это число. Ведущие нули в записи числа не допускаются.

Формат ввода: В единственной строке записаны целые числа a_0, a_1, \dots, a_9 ($0 \leq a_i \leq 1000$; $a_0 + a_1 + \dots + a_9 > 0$). Числа разделены пробелами.

Формат вывода: Выведите единственное целое положительное число без ведущих нулей — минимально возможное число, кратное 15, которое Петя может выложить из магнитов, используя их все. Если Петя не может выложить ни одного положительного числа, кратного 15, выведите -1.

Пример 1

input.txt:	output.txt:
1 1 0 0 0 1 0 0 0 0	105

Пример 2

input.txt:	output.txt:
1 0 0 0 0 1 0 0 0 0	-1

11.5. «Гуляя по столице». После того, как Петя Торопыжкин отдохнул в деревне у бабушки, он поехал с родителями в Москву. Благодаря стараниям мэра, столица обзавелась большим количеством кольцевых дорог. Эти дороги представляют собой окружности с центром в начале координат и радиусами, выражаемыми всевозможными целыми числами от 1 до 100 000.

Родители Пети наметили культурную программу — осмотр n достопримечательностей, расположенных в точках с координатами (x_i, y_i) . Они хотят добраться к первой достопримечательности на такси и затем перемещаться от одной достопримечательности к другой пешком по прямой. Им интересно, пересекает ли очередной отрезок их пути хотя бы одну кольцевую дорогу (касание также считается пересечением). Помогите родителям Пети — напишите программу, которая ответит на этот вопрос.

Формат ввода: В первой строке записано целое число n — количество точек маршрута ($2 \leq n \leq 100$). В каждой из n следующих строк записаны вещественные числа x_i, y_i , по модулю не превышающие 10 000 и заданные с точностью не более двух знаков после десятичной точки — координаты очередной точки маршрута. Никакие две соседние точки маршрута не совпадают. Числа разделены пробелами.

Формат вывода: Выведите $n - 1$ строку, в i -й из которых будет записано RISKY, если переход между i -й и $(i + 1)$ -й точками пересекает или касается хотя бы одной из кольцевых дорог, и SAFE в противном случае.

Пример

input.txt:	output.txt:
4	RISKY
2.2 1.91	RISKY
1 1	SAFE
1 -1	
1 -1.1	