

Муниципальное образование «Гурьевский городской округ»

Всероссийская олимпиада школьников по информатике
(школьный этап)

2017-2018 учебный год

11 класс

Максимальное количество баллов – 100

Время выполнения – 3 астрономических часа

Вводная часть

Каждому участнику школьного этапа предлагается для решения пять задач. Решением задачи является **исходный текст программы**.

Жюри рассматривает только **консольные приложения** на языках программирования **Паскаль, C/C++, Basic, java**. Жюри располагает компиляторами Pascal ABC.NET, Dev-Cpp, VB6, CodeBlocs. В комментариях участник может указать желаемый компилятор.

Программа должна вводить исходные данные из текстового файла **input.txt** и выводить результат в текстовый файл **output.txt**. Примеры файлового ввода и вывода приведены на следующей странице.

Файлы с исходными текстами программ необходимо называть **в точности так**, как указано в условии. Например, если участник пишет на языке C++, то файл с решением **задачи 1** он должен назвать **win.cpp** (если программа написана на другом языке, то расширение будет другим). Жюри не рассматривает неправильно названные файлы. Кроме того, участник **не должен сдавать несколько вариантов** решения одной и той же задачи. В этом случае жюри **может не рассматривать ни один из вариантов**.

Для каждой задачи жюри имеет по 20 тестов. Тесты, приведённые в условии задачи, могут входить или не входить в состав тестов жюри. Количество баллов, набранных участником, в точности равно количеству тестов, пройденных его программой. Тест считается пройденным, если программа вывела верный результат и работала не более 1 секунды. Таким образом, участник школьного этапа может набрать в сумме 100 баллов.

Задание 1 (20 баллов). Цифровой корень.**Имя файла с исходным текстом программы:****root.dpr | root.pas | root.cpp | root.c | root.bas | root.java***(Время: 1 сек. Память: 64 Мб Баллы: 20)*

Цифровой корень натурального числа получается следующим образом. Складываются все цифры данного числа. Процесс повторяется, пока в результате не будет получено однозначное число, которое и называется цифровым корнем числа.

Формат входных данных

На вход программе подаётся натуральное число $n \leq 10^6$.

Формат выходных данных

Выведите его цифровой корень.

Примеры входа и выхода

№	input.txt	output.txt
1	10	1
2	888	6

Задание 2 (20 баллов). Молоко и пирожок.**Имя файла с исходным текстом программы:****milk.dpr | milk.pas | milk.cpp | milk.c | milk.bas | milk.java***(Время: 1 сек. Память: 64 Мб Баллы: 20)*

Ученикам первого класса дополнительно дают стакан молока и пирожок, если вес первоклассника менее 30 кг. В первых классах школы учатся n учеников. Стакан молока имеет емкость 200 мл, а упаковки молока – 0,9 л. Определить количество дополнительных пакетов молока и пирожков, необходимых каждый день.

Формат входных данных

В первой строке задано целое число n ($0 < n \leq 1000$). В следующей строке идут n положительных действительных чисел – массы первоклассников.

Формат выходных данных

В одной строке вывести два целых числа - количество дополнительных пакетов молока и пирожков, необходимых каждый день.

Примеры входа и выхода

№	input.txt	output.txt
1	20 30 37 31 25 32 29 35 40 28 25 30 34 26 23 20 22 21 30 38 33	2 9

Задание 3 (20 баллов). Холодильник.

Имя файла с исходным текстом программы:

fridge.dpr | fridge.pas | fridge.cpp | fridge.c | fridge.bas | fridge.java

(Время: 1 сек. Память: 64 Мб Баллы: 20)

Холодильник должен быть параллелепипедом $A \times B \times C$ и хранить N кубических баночек мороженого размером $1 \times 1 \times 1$. Для уменьшения потерь холода, общая площадь поверхности холодильника должна быть как можно меньше. Например, если размер холодильника должен быть равен 12, то возможны следующие варианты:

Размеры холодильника	Площадь поверхности
$3 \times 2 \times 2$	32
$4 \times 3 \times 1$	38
$6 \times 2 \times 1$	40
$12 \times 1 \times 1$	50

По данному объему холодильника определите его наилучшую конфигурацию.

Формат входных данных

Программа получает на вход одно натуральное число N , не превосходящее 10^6 .

Формат выходных данных

Программа должна вывести три числа - оптимальные стороны холодильника в любом порядке. Если решений несколько, нужно вывести любое из них.

Примеры входа и выхода

№	input.txt	output.txt
1	12	3 2 2
2	13	1 13 1

Задание 4 (20 баллов). Шарики.

Имя файла с исходным текстом программы:

beads.dpr | beads.pas | beads.cpp | beads.c | beads.bas | beads.java

(Время: 1 сек. Память: 64 Мб Баллы: 20)

В одной компьютерной игре игрок выставляет в линию шарики разных цветов. Когда образуется непрерывная цепочка из трех и более шариков одного цвета, она удаляется из линии.

Напишите программу, которая по данной ситуации определяет, сколько шариков будет сейчас "уничтожено". Естественно, непрерывных цепочек из трех и более одноцветных шаров в начальный момент может быть более одной.

Формат входных данных

Сначала вводится количество шариков в цепочке (не более 10000) и цвета шариков (от 0 до 9, каждому цвету соответствует свое целое число).

Формат выходных данных

Требуется вывести количество шариков, которое будет "уничтожено".

Примеры входа и выхода

input.txt	output.txt
5 1 3 3 3 2	3
9 1 1 1 5 5 3 3 3 3	7

Задание 5 (20 баллов). Количество маршрутов кузнечика.

Имя файла с исходным текстом программы:

ways.dpr | ways.pas | ways.cpp | ways.c | ways.bas | ways.java

(Время: 1 сек. Память: 64 Мб Баллы: 20)

На числовой прямой в точке $x=1$ сидит кузнечик, который прыгает вправо (в направлении возрастания значения x), за один прыжок он может переместиться на 1, 2 или 3 единицы (то есть первым прыжком кузнечик может оказаться в точках $x=2$, $x=3$ или $x=4$).

Сколько существует различных маршрутов кузнечика, начинающихся в точке 1 и заканчивающихся в точке $x=N$?

Формат входных данных

Во входном файле **input.txt** задано единственное число N ($1 \leq N \leq 50$).

Формат выходных данных

В выходной файл **output.txt** нужно вывести одно число – количество различных маршрутов кузнечика.

Примеры входа и выхода

№	input.txt	output.txt
1	4	4
2	30	29249425

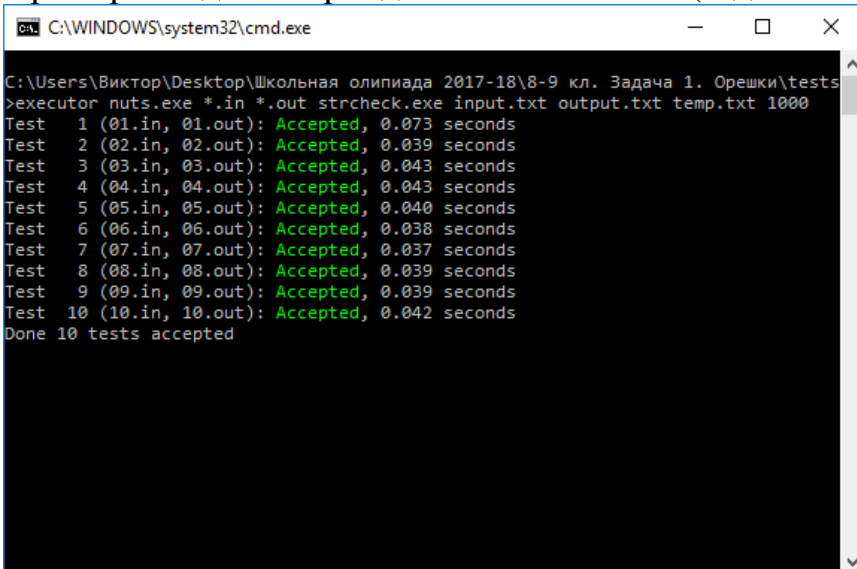
Проверка олимпиады.

Задания школьного этапа олимпиады по информатике (программированию) выполняются на компьютере.

Учащиеся должны сдавать на проверку исходные тексты программ, которые члены жюри должны откомпилировать и запустить на выполнение. Члены жюри не могут вносить изменения в тексты сданных решений, поэтому учащиеся должны откомпилировать и проверить решения на своих компьютерах.

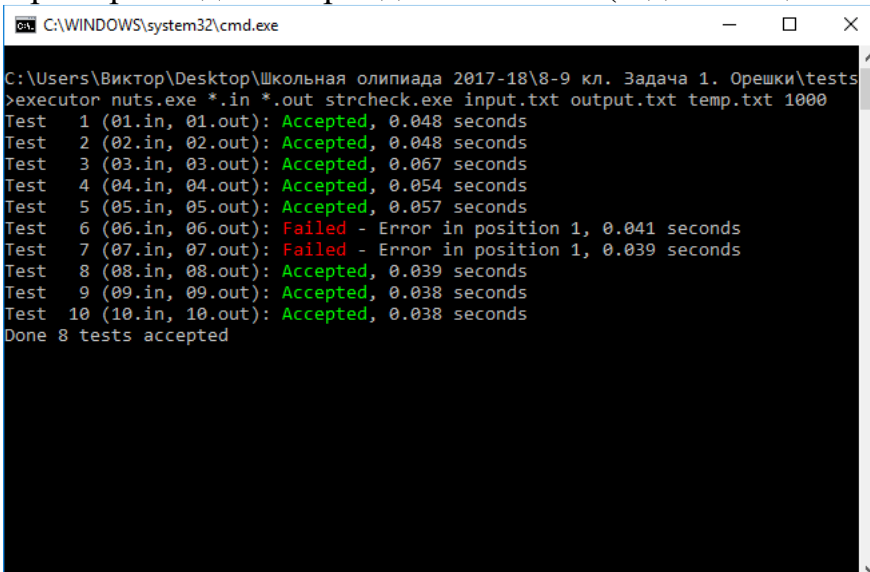
Для удобства проверки в каждой папке задачи есть тесты, которые расположены в папке «tests» там же находится программа «Executor.exe» - для автоматической проверки решения задачи. Для запуска проверки необходимо откомпилированное решение, с расширением «.exe», поместить в папку «tests» и запустить файл «test.bat». По окончании работы Вы увидите результат проверки. Напротив каждого теста должно стоять слово «Accepted» – принято. В конце будет строка с общим количеством удачно пройденных тестов.

Пример 1. Удачно пройдены все 10 тестов (задание оценивается в 10 баллов).



```
C:\WINDOWS\system32\cmd.exe
C:\Users\Виктор\Desktop\Школьная олимпиада 2017-18\8-9 кл. Задача 1. Орешки\tests
>executor nuts.exe *.in *.out strcheck.exe input.txt output.txt temp.txt 1000
Test 1 (01.in, 01.out): Accepted, 0.073 seconds
Test 2 (02.in, 02.out): Accepted, 0.039 seconds
Test 3 (03.in, 03.out): Accepted, 0.043 seconds
Test 4 (04.in, 04.out): Accepted, 0.043 seconds
Test 5 (05.in, 05.out): Accepted, 0.040 seconds
Test 6 (06.in, 06.out): Accepted, 0.038 seconds
Test 7 (07.in, 07.out): Accepted, 0.037 seconds
Test 8 (08.in, 08.out): Accepted, 0.039 seconds
Test 9 (09.in, 09.out): Accepted, 0.039 seconds
Test 10 (10.in, 10.out): Accepted, 0.042 seconds
Done 10 tests accepted
```

Пример 2. Удачно пройдено 8 тестов (задание оценивается в 8 баллов).



```
C:\WINDOWS\system32\cmd.exe
C:\Users\Виктор\Desktop\Школьная олимпиада 2017-18\8-9 кл. Задача 1. Орешки\tests
>executor nuts.exe *.in *.out strcheck.exe input.txt output.txt temp.txt 1000
Test 1 (01.in, 01.out): Accepted, 0.048 seconds
Test 2 (02.in, 02.out): Accepted, 0.048 seconds
Test 3 (03.in, 03.out): Accepted, 0.067 seconds
Test 4 (04.in, 04.out): Accepted, 0.054 seconds
Test 5 (05.in, 05.out): Accepted, 0.057 seconds
Test 6 (06.in, 06.out): Failed - Error in position 1, 0.041 seconds
Test 7 (07.in, 07.out): Failed - Error in position 1, 0.039 seconds
Test 8 (08.in, 08.out): Accepted, 0.039 seconds
Test 9 (09.in, 09.out): Accepted, 0.038 seconds
Test 10 (10.in, 10.out): Accepted, 0.038 seconds
Done 8 tests accepted
```