



**Школьный этап
всероссийской олимпиады школьников
по информатике**
Брянский район

Разработал:
Жигулов Кирилл Александрович
учитель информатики
МБОУ «Новодарковичская СОШ»

Брянск 2020

Задания для 7-8 классов

Задача 1. Торт

На свой день рождения Петя купил красивый и вкусный торт, который имел идеально круглую форму. Петя не знал, сколько гостей придет на его день рождения, поэтому вынужден был разработать алгоритм, согласно которому он сможет быстро разрезать торт на N равных частей. Следует учесть, что разрезы торта можно производить как по радиусу, так и по диаметру.

Помогите Пете решить эту задачу, определив наименьшее число разрезов торта по заданному числу гостей.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – число гостей, включая самого виновника торжества ($N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимально возможное число разрезов торта.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	3	3

Задача 2. Три толстяка

Три толстяка решили поспорить: кто из них самый тяжелый. После взвешивания оказалось, что их масса соответственно M_1 , M_2 и M_3 килограмм. Считается, что масса толстяка должна быть не менее 94 и не более 727 килограмм.

Помогите определить массу самого тяжелого из них, либо выяснить, что была допущена ошибка при взвешивании.

Входные данные

Входной файл INPUT.TXT содержит три целых числа M_1 , M_2 и M_3 , разделенные пробелом. Все числа целые и не превосходят 10 000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT выведите массу самого тяжелого толстяка в случае корректного взвешивания, либо слово «Error» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	98 106 214	214
2	100 100 100	100
3	93 500 1000	Error

Задача 3. Клавиатура

Для данной буквы английского алфавита нужно вывести справа стоящую букву на стандартной клавиатуре. При этом клавиатура замкнута, т.е. справа от буквы «р» стоит буква «а», от буквы «l» стоит буква «z», а от буквы «m» — буква «q».

Входные данные

Первая строка входного файла INPUT.TXT содержит один символ — маленькую букву английского алфавита.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести букву, стоящую справа от заданной буквы, с учетом замкнутости клавиатуры.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	q	w
2	t	y
3	p	a
4	l	z
5	m	q

Задача 4. Перепись

В доме живет N жильцов. Однажды решили провести перепись всех жильцов данного дома и составили список, в котором указали возраст и пол каждого жильца. Требуется найти номер самого старшего жителя мужского пола.

Входные данные

Во входном файле INPUT.TXT в первой строке задано натуральное число N — количество жильцов ($N \leq 100$). В последующих N строках располагается информация о всех жильцах: каждая строка содержит два целых числа: V и S — возраст и пол человека ($1 \leq V \leq 100$, $S = 0$ или 1). Мужскому полу соответствует значение $S=1$, а женскому — $S=0$.

Выходные данные

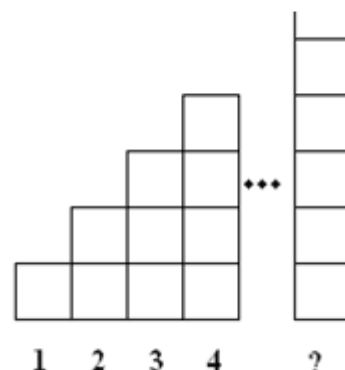
Выходной файл OUTPUT.TXT должен содержать номер самого старшего мужчины в списке. Если таких жильцов несколько, то следует вывести наименьший номер. Если жильцов мужского пола нет, то выведите -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 25 1 70 1 100 0 3 1	2
2	2 25 0 25 1	2

Задача 5. Лестница

Мальчик Петя строит из кубиков лестницу. Лестница представляет собой несколько строящихся рядом башенок из кубиков, каждая из которых ровно на один кубик выше предыдущей. Требуется по имеющемуся у мальчика Пети числу кубиков определить, какой в кубиках будет высота последней ступеньки.



Входные данные

Входной файл INPUT.TXT содержит число K – количество кубиков у мальчика Пети ($1 \leq K \leq 10^6$).

Выходные данные

В выходной файл OUTPUT.TXT выведите количество кубиков в последней ступеньке у максимально высокой лестницы, которую можно построить из K кубиков.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	4	2
3	6	3

Рекомендации для проверяющих

Представленные в олимпиаде задачи взяты из открытого интернет-источника «Школа программиста» (www.acmp.ru).

1. Торт: https://acmp.ru/index.asp?main=task&id_task=539
2. Три толстяка: https://acmp.ru/index.asp?main=task&id_task=754
3. Клавиатура: https://acmp.ru/index.asp?main=task&id_task=66
4. Перепись: https://acmp.ru/index.asp?main=task&id_task=131
5. Лестница: https://acmp.ru/index.asp?main=task&id_task=514

Рекомендуется проведение онлайн-олимпиады с использованием данного ресурса, так как работа с ним максимально похожа на способ проведения олимпиад следующих уровней (муниципального и регионального). Для такого варианта проведения олимпиады необходима предварительная регистрация участников на сайте. Система сайта обеспечивает автоматическую проверку решений. В таком случае, за каждое задание участник олимпиады получает количество баллов, равное количеству успешно пройденных тестов.

Максимальное количество баллов по задачам:

- | | |
|------------------|------------|
| 1. Торт: | 20 баллов* |
| 2. Три толстяка: | 20 баллов |
| 3. Клавиатура: | 26 баллов |
| 4. Перепись: | 18 баллов* |
| 5. Лестница: | 32 баллов* |

Тест	Результат	Время	Память
1	Accepted	0,015	584 Кб
2	Accepted	0,015	584 Кб
3	Accepted	0,015	584 Кб
4	Accepted	0,015	584 Кб
5	Accepted	0,015	584 Кб
6	Accepted	0,015	580 Кб
7	Wrong answer	0,015	580 Кб

Посмотреть количество набранных баллов можно кликнув на ID номер решения. Например, в случае на рисунке выше, участник получает 6 баллов, так как успешно пройдено только 6 тестов.

** С целью уравнивания «веса» заданий, в задаче «Торт» 2 балла дается за 1 верный тест, в задаче «Перепись» 3 балла дается за 1 верный тест, в задаче «Лестница» 2 балла дается за 1 верный тест.*

Второй вариант проведения олимпиады – без использования автоматизированной системы проверки. Участники пишут программы и сдают их проверяющему в электронном (желательно) или письменном виде. В таком случае проверяющий должен запустить каждую программу и проверить ее работоспособность на 4 тестах, предложенных в комментариях к решению. При верном ответе участник олимпиады получает по 6 баллов каждый тест. Таким образом максимально в сумме можно набрать $5 \cdot 4 \cdot 6 = 120$ баллов, что примерно равно максимуму онлайн-версии (116 баллов).

Рекомендуемый допуск на муниципальный этап: 60 баллов (50% выполнения работы) либо три полностью решенных задачи.

Разбор заданий

Задача 1. Торт. (Сложность 6%)

Если количество гостей чётно, то торт можно разрезать по диаметру, количество разрезов равно $n/2$; если нечётно – по радиусу, тогда количество разрезов равно n . Частный случай, если гостей не пришло ($n=1$). В таком случае разрезать торт не нужно.

№ теста	Ввод	Вывод
1	12	6
2	888	444
3	993	993
4	1	0
Вариант решения на Pascal:		
<pre> var n:integer; begin read(n); if n>1 then if odd (n) then write (n) else write (n div 2) else write(0); end.</pre>		

Задача 2. Три толстяка. (Сложность 7%)

Найти максимальное число из трех и проверить, чтобы введенные числа были в заданном диапазоне.

№ теста	Ввод	Вывод
1	200 400 300	400
2	555 444 555	555
3	777 777 777	Error
4	123 89 321	Error
Вариант решения на Pascal:		
<pre> var a,b,c,max:integer; begin read(a,b,c); max:=a; if b>max then max:=b; if c>max then max:=c; if (a>727) or (a<94) or (b>727) or (b<94) or (c>727) or (c<94) then write ('Error') else write (max); end.</pre>		

Задача 3. Клавиатура. (Сложность 11%)

Работа с символами. Можно решить «в лоб», используя 26 условных операторов. Более «элегантное» решение – записать последовательность букв в строку и выводить следующую по счету – предложено ниже.

№ теста	Ввод	Вывод
1	p	a
2	l	z
3	h	j
4	m	q
Вариант решения на Pascal:		
<pre> var n:char; s:string; begin read (n); s:='qwertyuiopasdfghjklzxcvbnmq'; write (copy(s,pos(n,s)+1,1)); end.</pre>		

Задача 4. Перепись. (Сложность 15%)

Поиск максимума в последовательности с дополнительным условием. Комбинация цикла с параметром и сложного ветвления.

№ теста	Ввод	Вывод
1	3 22 1 33 1 44 0	2
2	4 99 0 66 1 88 1 77 0	3
3	5 44 1 60 0 50 0 55 1 55 1	4
4	2 22 0 33 0	-1
Вариант решения на Pascal:		

```

var n,v,s,k,max,i:integer;
begin
  read (n);
  k:=-1; max:=0;
  for i:=1 to n do begin
    read (v,s);
    if (s=1) and (v>max) then
      begin max:=v; k:=i; end;
    end;
  write (k);
  end.

```

Задача 5. Лестница. (Сложность 21%)

Несложное математическое моделирование можно реализовать с помощью цикла с предусловием. До тех пор, пока хватает кубиков на нужную высоту, отбрасывать каждый раз увеличивающееся на один число кубиков.

№ теста	Ввод	Вывод
1	11	4
2	200	19
3	5050	100
4	999888	1413

Вариант решения на Pascal:

```

var n,i:integer;
begin
  read(n);
  i:=0;
  while n>i do begin
    i:=i+1;
    n:=n-i;
  end;
  write (i)
end.

```

Задания для 9-11 классов

Задача 1. Бинарные числа

Говорят, что плохой программист – это тот, кто считает, что в одном килобайте 1000 байт, а хороший программист – это тот, кто полагает, что в одном километре 1024 метра.

Многим эта шутка понятна, так как все знают, что в процессах, связанных с информатикой и компьютерной техникой, фигурирует множество значений, выражаемых степенью двойки, то есть чисел вида 2^K , где K – некоторое неотрицательное целое число. Назовем такие числа *бинарными*. Это такие числа как 2, 4, 8, 16, 32 и т.д. Действительно, когда речь идет о размере памяти или о разрешении экрана монитора, то мы часто наталкиваемся на бинарные числа. Все это связано с принципом хранения информации в памяти ЭВМ.

Задано целое число N . Требуется определить, является ли оно бинарным.

Входные данные

Входной файл INPUT.TXT содержит единственное целое число N , не превосходящее 10000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если заданное число является бинарным, и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1024	YES
2	23	NO

Задача 2. Орфография

У студента-филолога Васи есть замечательный друг Петя. И Петя никак не может выучить английский язык. Английский текст Петя еще кое-как читает, но пишет с ужасными ошибками, причем чаще всего он вставляет в слова лишние буквы.

Вася решил помочь Пете. Теперь каждый день Вася диктует Пете слова, а Петя их записывает. После семестра занятий Петя стал писать много лучше и теперь делает в словах только по одной ошибке. Чтобы автоматизировать процесс исправления ошибок, Вася просит Вас написать программу, которая удаляет из слова одну лишнюю букву и показывает Пете правильное слово.

Входные данные

Входной файл INPUT.TXT содержит целое число K - номер лишней буквы, а затем через один или несколько пробелов записано слово S , состоящее из английских букв верхнего регистра. Гарантируется, что номер буквы не превышает длину слова. Длина слова не более 80 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите исправленное слово.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 MISTSPELL	MISSPELL
2	2 ABC	AC

Задача 3. Напёрстки

Шулер показывает следующий трюк. Он имеет три одинаковых наперстка. Под первый (левый) он кладет маленький шарик. Затем он очень быстро выполняет ряд перемещений наперстков, каждое из которых – это одно из трех перемещений - А, В, С:

- А - обменять местами левый и центральный наперстки,
- В - обменять местами правый и центральный наперстки,
- С - обменять местами левый и правый наперстки.

Необходимо определить, под каким из наперстков окажется шарик после всех перемещений.

Входные данные

В единственной строке входного файла INPUT.TXT записана строка длиной от 1 до 50 символов из множества $\{A, B, C\}$ – последовательность перемещений.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести номер наперстка, под которым окажется шарик после перемещений.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	СВАВСАССС	1

Задача 4. Клавиатура-2

Всем известно, что со временем клавиатура изнашивается, и клавиши на ней начинают залипать. Конечно, некоторое время такую клавиатуру еще можно использовать, но для нажатий клавиш приходится использовать большую силу.

При изготовлении клавиатуры изначально для каждой клавиши задается количество нажатий, которое она должна выдерживать. Если знать эти величины для используемой клавиатуры, то для определенной последовательности нажатых клавиш можно определить, какие клавиши в процессе их использования сломаются, а какие – нет.

Требуется написать программу, определяющую, какие клавиши сломаются в процессе заданного варианта эксплуатации клавиатуры.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N ($1 \leq N \leq 100$) – количество клавиш на клавиатуре. Вторая строка содержит n целых чисел – $c_1, c_2, \dots, c_N$, где c_i ($1 \leq c_i \leq 100000$) – количество нажатий, выдерживаемых i -ой клавишей. Третья строка содержит целое число K ($1 \leq K \leq 100000$) – общее количество нажатий клавиш, и последняя строка содержит K целых чисел p_j ($1 \leq p_j \leq N$) – последовательность нажатых клавиш.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести N строк, содержащих информацию об исправности клавиш. Если i -ая клавиша сломалась, то i -ая строка должна содержать слово "yes" (без кавычек), если же клавиша работоспособна – слово "no".

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 50 3 4 3 16 1 2 3 4 5 1 3 3 4 5 5 5 5 4 5	yes no no no yes

Задача 5. Снова Фибоначчи

Вам наверняка знакомы числа Фибоначчи: 1, 1, 2, 3, 5, 8, 13, 21... Они определяются рекуррентным соотношением: $F_n = F_{n-1} + F_{n-2}$, $F_0 = F_1 = 1$.

Требуется найти последнюю цифру n -го числа Фибоначчи.

Входные данные

Во входном файле INPUT.TXT содержится одно целое число n ($0 \leq n \leq 10^8$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число - последнюю цифру числа F_n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	5	8

Рекомендации для проверяющих

Представленные в олимпиаде задачи взяты из открытого интернет-источника «Школа программиста» (www.acmp.ru).

1. Бинарные числа: https://acmp.ru/index.asp?main=task&id_task=692
2. Орфография: https://acmp.ru/index.asp?main=task&id_task=940
3. Напёрстки: https://acmp.ru/index.asp?main=task&id_task=678
4. Клавиатура-2: https://acmp.ru/index.asp?main=task&id_task=534
5. Снова Фибоначчи: https://acmp.ru/index.asp?main=task&id_task=623

Рекомендуется проведение онлайн-олимпиады с использованием данного ресурса, так как работа с ним максимально похожа на способ проведения олимпиад следующих уровней (муниципального и регионального). Для такого варианта проведения олимпиады необходима предварительная регистрация участников на сайте. Система сайта обеспечивает автоматическую проверку решений. В таком случае, за каждое задание участник олимпиады получает количество баллов, равное количеству успешно пройденных тестов.

Максимальное количество баллов по задачам:

- | | |
|---------------------|------------|
| 1. Бинарные числа: | 25 баллов |
| 2. Орфография: | 15 баллов* |
| 3. Напёрстки: | 20 баллов* |
| 4. Клавиатура-2: | 21 балл |
| 5. Снова Фибоначчи: | 30 баллов* |

Тест	Результат	Время	Память
1	Accepted	0,015	584 Кб
2	Accepted	0,015	584 Кб
3	Accepted	0,015	584 Кб
4	Accepted	0,015	584 Кб
5	Accepted	0,015	584 Кб
6	Accepted	0,015	580 Кб
7	Wrong answer	0,015	580 Кб

Посмотреть количество набранных баллов можно кликнув на ID номер решения. Например, в случае на рисунке выше, участник получает 6 баллов, так как успешно пройдено только 6 тестов.

** С целью уравнивания «веса» заданий, в задаче «Орфография» 1 балл даётся за 2 верных теста, в задаче «Напёрстки» 2 балла даётся за 1 верный тест, в задаче «Снова Фибоначчи» 2 балла даётся за 1 верный тест.*

Второй вариант проведения олимпиады – без использования автоматизированной системы проверки. Участники пишут программы и сдают их проверяющему в электронном (желательно) или письменном виде. В таком случае проверяющий должен запустить каждую программу и проверить ее работоспособность на 4 тестах, предложенных в комментариях к решению. При верном ответе участник олимпиады получает по 6 баллов за каждый тест. Таким образом максимально в сумме можно набрать $5 \cdot 4 \cdot 6 = 120$ баллов, что примерно равно максимуму онлайн-версии (111 баллов).

Рекомендуемый допуск на муниципальный этап: 60 баллов (50% выполнения работы) либо три полностью решенных задачи.

Разбор заданий

Задача 1. Бинарные числа (Сложность 8%)

Цикл с предусловием и ветвление. Единица – особый случай, который не упоминается в условии, но $2^0=1$, поэтому единица – бинарное число. Решения, постоянно выводящие «YES» или «NO» (независимо от входных данных), оцениваются в 0 баллов.

№ теста	Ввод	Вывод
1	5555	NO
2	256	YES
3	8192	YES
4	1	YES
Вариант решения на Pascal:		
<pre> var n,k:integer; begin readln(n); k:=1; while n>k do k:=k*2; if n=k then write ('YES') else write ('NO'); end. </pre>		

Задача 2. Орфография. (Сложность 13%)

Задача на разбор строки. Необходимо отделить число от строки, а затем удалить пробелы и символ с нужным номером.

№ теста	Ввод	Вывод
1	6 INFORMSMATIKA	INFORMATIKA
2	10 OLIMPIADAP	OLIMPIADA
3	1 URA	RA
4	24 ABCDEFGHIJKLMNOPQRSTUVWXYZ	ABCDEFGHIJKLMNOPQRSTUVWXYZ
Вариант решения на Pascal:		
<pre> var t, n, x: integer; s, c: string; begin readln(s); n := pos(' ', s); c := copy(s, 1, n - 1); val(c, x, t); delete(s, 1, n); while s[1] = ' ' do delete(s, 1, 1); delete(s, x, 1); writeln(s); end. </pre>		

Задача 3. Напёрстки. (Сложность 15%)

Вариант решения на Pascal:

```

var n,i,k,j,q:integer;
b,c:array [1..100] of longint;
begin
  read(n);
  for i:=1 to n do begin read (c[i]); b[i]:=0; end;
  read (k);
  for i:=1 to k do begin read (q); inc(b[q]); end;
  for i:=1 to n do
    if b[i]>c[i] then writeln ('yes') else writeln ('no');
  end.

```

Задача 5. Снова Фибоначчи. (Сложность 23%)

Хитрость задачи в том, что, получив очередное число Фибоначчи, нам нужна только его последняя цифра, иначе числа выйдут за предел целочисленных форматов. В остальном, это классическая задача на поиск чисел Фибоначчи. Дополнительно необходимо учесть частный случай для двух первых чисел.

№ теста	Ввод	Вывод
1	100	1
2	1234	5
3	55555	7
4	7654321	1

Вариант решения на Pascal:

```

var n,i,f0,f1,f:integer;
begin
  read (n);
  f0:=1;
  f1:=1;
  for i:=2 to n do
    begin
      f:=(f0+f1) mod 10;
      f0:=f1;
      f1:=f;
    end;
  if n>1 then write (f) else write (1);
end.

```

Уважаемые коллеги, отобранные задания имеют невысокую сложность, их можно отнести к базовому уровню. Заметно, что программы состоят всего из нескольких строчек, и не содержат элементов, не изучаемых в школьном курсе информатики в теме «алгоритмы и программирование». Суть олимпиадных заданий по информатике (программированию) состоит в умении **понять условие задачи** и реализовать алгоритм решения на известном языке программирования. Опыт в решении подобных задач приходит только с практикой.

Если вам кажется, что данные задания сложны для учащихся 9-11 классов вашей школы, допускается использование комплекта задач для 7-8 классов, с соответствующими критериями.