

## Отборочный тур для 3-5 классов

### Т1. Миньоны

90 Миньонов выстроились в шеренгу с интервалом в 2 метра. Грю собирается пройти от первого до последнего Миньона и каждому пожать руку. Сколько метров ему придется пройти?

**Ответ:** 178. **Решение.** Между 90 Миньонами 89 интервалов. Так как каждый из них равен 2 метрам, то всего Грю пройдет  $89 \cdot 2 = 178$ .

### Т2. Интересное число

Найдите наименьшее натуральное число кратное 5, сумма цифр которого равна 55.

**Ответ:** 5999995. **Решение.** Для того, чтобы число было наименьшим, нам нужно использовать как можно меньше цифр. Чтобы набрать сумму цифр 55 как можно меньшим количеством цифр, нужно использовать как можно большие цифры. Девятками мы можем набрать максимум 45. Чтобы число было кратно 5, нужно чтобы оно оканчивалось на 0 или 5. С нулем на конце наименьшим будет 19999990, а с пятеркой на конце 5999995. Меньшее из этих двух чисел и является ответом.

### Т3. Много ли чисел

Сколько существует двузначных чисел, у которых цифра десятков больше цифры единиц?

**Ответ:** 45. **Решение.** Попробуем перебрать такие числа. С единицей в числе десятков такое число одно, с двойкой — два, с тройкой — три, ..., с девяткой — девять. Сумма цифр от одного до девяти равна 45.

### Т4. Угадывание числа

Специальный агент Дейл Купер и Даяна играют в такую игру:

Даяна загадывает число от 1 до 356, а Агент Купер, задавая вопросы, на которые можно ответить только «ДА» или «НЕТ», должен отгадать число Даяны. За какое наименьшее количество вопросов Специальный Агент гарантировано сможет угадать число Даяны?

**Ответ:** 9. **Решение.** Воспользуемся методом двоичного поиска. Каждым вопросом мы можем отсекаем половину чисел. Для этого первым вопросом будет: “Больше ли загаданное число, чем  $365 // 2 = 182$  ?” ( $//$  -- целочисленное деление). Затем благодаря ответу мы определим в каком промежутке находится загаданное число: от 1 до 182 или от 183 до 365. Полученный промежуток поделим пополам и зададим аналогичный вопрос и так далее. Всего нам потребуется 9 вопросов, так как  $2^9 = 512 > 365$ .

## Т5. Шарик и Матроскин

Шарик попросил Кота Матроскина написать как можно больше двузначных чисел так, чтобы среди них не было двух, дающих в сумме 100. Какое наибольшее количество чисел может написать Матроскин?

**Ответ: 50. Решение.** Для каждого двузначного числа  $n$  его парой, дающей в сумме 100 является число  $100-n$ . Однозначные числа от 1 до 9 составляют пары числам от 99 до 91, следовательно их точно можно написать. Из оставшегося 81 двузначного числа можно максимум выписать 41. Проиллюстрируем это: все числа, кроме 50 можно разбить на пары, дающие в сумме 100. Из каждой такой пары можем выбрать не более одного числа. Следовательно, всего можно выписать максимум 50 чисел.

## Т6. 2018-ый член последовательности

Найдите 2018-й член последовательности: 2, 6, 12, 20, 30, ... .

**Ответ: 4074342. Решение.** Заметим, что  $n$ -ый член последовательности находится по формуле  $n \cdot (n + 1)$ . Следовательно, 2018-ый член последовательности равен  $2018 \cdot 2019 = 4074342$ .

## Т7. Кащей и Василиса

У Кащея Бессмертного всего 169 монет, причем одна из них – легкая фальшивая монета. У Василисы Премудрой есть весы, но за каждое взвешивание Кощей должен заплатить. Причем, если одна из чашек перевесила, Кащей заплатит 1 золотой, а если весы остались в равновесии, то 2 золотых. Какую наибольшую сумму может заработать Василиса, если Кащей твердо намерен узнать, какая монета фальшивая?

**Ответ: 6. Решение.** Назовем  $T_n$  -- число монет, из которых Кащей точно сможет определить фальшивую, затратив при этом не больше  $n$  золотых. В таком случае, на каждую из чаш можно положить не более  $T_{n-1}$  монет (так как в случае неравенства, Кащею нужно будет заплатить 1 золотой, а затем определять фальшивую монету из  $T_{n-1}$  монет), а вне чашек может остаться соответственно не больше  $T_{n-2}$  монет. Значит, можем составить такое неравенство:  $T_n \leq T_{n-2} + 2T_{n-1}$

Очевидно, что  $T_0 = 1$  (если ничего не взвешивать, то фальшивая монета должна быть единственной),  $T_1 = 2$ . Затем, можем последовательно найти, что  $T_2 \leq 5, T_3 \leq 12, T_4 \leq 29, T_5 \leq 70, T_6 \leq 169$

Покажем, что 6 золотых нам хватит. Положим на две чаши весов по 70 монет и 29 монет отложим. Тогда если весы окажутся в равновесии, то за 4 оставшихся золотых, мы сможем найти фальшивую монету среди 29 монет, и т.д., а если одна из чаш перевесит, то за 5 оставшихся золотых мы сможем найти фальшивую монету среди 70 монет и т.д.

## T8. Аня ищет числа

У Кащей Бессмертного всего 169 монет, причем одна из них – легкая фальшивая монета. У Василисы Премудрой есть весы, но за каждое взвешивание Кощей должен заплатить. Причем, если одна из чашек перевесила, Кащей заплатит 1 золотой, а если весы остались в равновесии, то 2 золотых. Какую наибольшую сумму может заработать Василиса, если Кащей твердо намерен узнать, какая монета фальшивая?

**Ответ: 15. Решение.** Запишем условие как:  $7 \cdot (10a + b) = 100a + b$ . Получим:

$$70a + 7b = 100a + b$$

$$6b = 30a$$

$$b = 5a$$

А так как  $a$  и  $b$  - цифры, то  $b = 5$ ,  $a = 1$ .

## P1. Чистка зубов

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

По четным дням месяца Вася чистит зубы, а по нечетным — не чистит. Чтобы не запутаться, он просит вас написать программу, которая по номеру дня говорит, нужно ему сегодня чистить зубы или нет.

Пн	Вт	Ср	Чт	Пт	Сб	Вс
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

### Формат ввода

Ввод содержит одно число  $n$  ( $1 \leq n \leq 31$ ).

### Формат вывода

Выведите YES, если сегодня надо чистить зубы или NO, если не надо.

### Пример 1

Ввод	Вывод
4	YES

### Пример 2

Ввод	Вывод
15	NO

### Пример 3

Ввод	Вывод
1	NO

### Решение.

В задаче нужно было проверить, что число во вводе четное. Для этого нужно взять остаток от деления этого числа на 2 и проверить, что он равен нулю.

Пример решения на Python:

```
n = int(input())
if n % 2 == 0:
    print("Yes")
else:
    print("No")
```

## P2. Дверь для Кроша

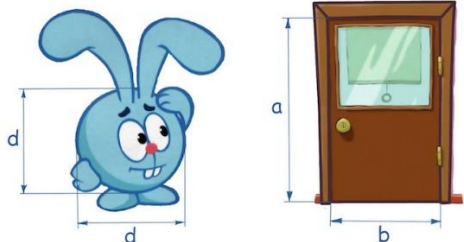
Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Крош построил новый дом и уже после постройки задумался, а достаточно ли большая в доме дверь? Помогите Крошу понять, смогут ли смешарики пройти через дверь. Для простоты будем считать, что каждый смешарик имеет форму шара с диаметром  $d$ , а дверь имеет форму прямоугольника  $a \times b$ .



## Формат ввода

На вход подаются целые положительные числа  $a$ ,  $b$  и  $d$ . Все числа не больше 1000.

## Формат вывода

Выведите строку «Yes», если Смешарики могут пройти в дверь и строку «No», если нет.

### Пример 1

Ввод	Вывод
------	-------

5 7 4	Yes
-------	-----

### Пример 2

Ввод	Вывод
------	-------

2 3 5	No
-------	----

### Пример 3

Ввод	Вывод
------	-------

6 6 6	Yes
-------	-----

## Решение.

В этой задаче нужно было проверить, что круг можно поместить в прямоугольник. Для этого нужно проверить, что диаметр круга не больше каждой из сторон прямоугольника.

Пример решения на Python:

```
a, b, r = map(int, input().split())
if r <= min(a, b):
    print("Yes")
else:
    print("No")
```

## Р9. Конфетки

Ограничение времени 1 секунда

Ограничение памяти 64Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

У Коли день рождения, и он хочет подарить всем  $n$  одноклассникам по конфетке. Конфеты в магазине продают упаковками по  $s$  штук, и каждая пачка стоит  $p$  рублей. Помогите Коле посчитать, сколько денег ему нужно потратить.

Например, если у Коли 21 друг, в упаковке 5 конфет, и каждая упаковка стоит 10 рублей, то Коле нужно купить 5 упаковок и потратить 50 рублей.

### Формат ввода

Ввод содержит три целых числа  $n$ ,  $s$  и  $p$ . Все числа от 1 до 100.

### Формат вывода

Выведите одно число — сумму, которую нужно потратить Коле.

### Пример 1

Ввод	Вывод
------	-------

21 5 10	50
---------	----

### Пример 2

Ввод	Вывод
------	-------

1 10 20	20
---------	----

### Пример 3

Ввод	Вывод
------	-------

37 2 12	228
---------	-----

### Решение.

Сначала найдем, сколько коробок конфет нужно купить. Для этого нужно поделить  $n$  на  $s$ , округлив ответ вверх. Это можно сделать разными способами, проще всего использовать формулу  $(n + s - 1) // s$  (где “//” — деление с округлением вниз). Далее умножим полученное число коробок на цену одной коробки.

Пример решения на Python:

```
n, c, p = map(int, input().split())
print((n + c - 1) // c * p)
```

## Р4. Коллекция покемонов

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Паша собирает наклейки с покемонами. Всего в его коллекции  $p$  наклеек, но на некоторых из них нарисованы одинаковые покемоны. Помогите ему посчитать, сколько разных покемонов у него есть. Для удобства он занумеровал всех известных ему покемонов числами от 1 до 100 и выписал для каждой наклейки в своей коллекции номер покемона, который на ней изображен.



### Формат ввода

Первая строка содержит число  $n$  ( $0 \leq n \leq 1000$ ). Вторая строка содержит  $n$  чисел — номера покемонов на наклейках из коллекции.

### Формат вывода

Выведите число различных покемонов в коллекции Паши.

### Пример 1

**Ввод**

```
6
5 4 4 3 5 1
```

**Вывод**

```
4
```

### Пример 2

**Ввод**

```
1
100
```

**Вывод**

```
1
```

**Решение.**

В этой задаче нужно было найти число различных элементов в массиве.

Есть много способов решить эту задачу. Например, можно отсортировать массив, тогда одинаковые элементы будут рядом. Теперь посчитаем только те элементы, которые не равны стоящему слева от них.

```
n = int(input())
a = list(map(int, input().split()))
a.sort()
c = 0
for i in range(n):
    if i == 0 or a[i - 1] != a[i]:
        c += 1
print(c)
```

Другой способ — использовать структуру данных **set** для хранения множества чисел, которые есть в массиве

```
n = int(input())
a = list(map(int, input().split()))
c = set()
for i in range(n):
    c.add(a[i])
print(len(c))
```

## P5. Секретное сообщение

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Принцесса Лея хочет послать секретное сообщение Люку, спрятав его в обычном письме. Чтобы расшифровать сообщение, нужно взять первую букву от каждого слова письма и сложить их вместе в одно слово. Помогите Люку расшифровать сообщение Леи.

### Формат ввода

Ввод содержит одну строку, состоящую из нескольких слов, разделенных пробелами. Каждое слово состоит из маленьких латинских букв. Длина письма не превышает 100 символов.



## Формат вывода

Выведите расшифрованное сообщение.

### Пример 1

#### Ввод

```
hear every little potato
```

#### Вывод

```
help
```

### Пример 2

#### Ввод

```
keep education knowledge
```

#### Вывод

```
kek
```

## Решение.

В этой задаче нужно взять первую букву из каждого слова в сообщении и соединить их в одно слово. Для того, чтобы разделить ввод на отдельные слова, в Python можно использовать операцию `split()`. Далее нужно пройтись, по всем словам, и выписать их первые буквы.

```
a = input().split()
for s in a:
    print(s[0], end="")
```

## Р6. Будильник

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Петя каждый вечер ставит будильник, чтобы не проспать, но он не всегда просыпается с первого раза, поэтому он купил будильник, который продолжает звонить каждые 9 минут до тех пор, пока Петя окончательно не проснется. Чтобы спланировать утро, Петя хочет узнать, когда будильник сделает первые 10 звонков. Учтите, что у Пети может быть странный режим дня и утро может начинаться в самое неожиданное время.

## Формат ввода

Ввод содержит два числа  $h$  ( $0 \leq h \leq 23$ ) и  $m$  ( $0 \leq h \leq 59$ ): время (часы и минуты), на которое установлен будильник.

## Формат вывода

Выведите время (часы и минуты) первых 10 звонков будильника.

### Пример 1

Ввод	Вывод
8 0	8 0
	8 9
	8 18
	8 27
	8 36
	8 45
	8 54
	9 3
	9 12
	9 21

### Пример 2

Ввод	Вывод
23 30	23 30
	23 39
	23 48
	23 57
	0 6
	0 15
	0 24
	0 33
	0 42
	0 51

## Решение.

В этой задаче нужно научиться вычислять, какое время будет через 9 минут. Для этого увеличим текущее число минут на 9, если оно стало больше или равно 60, то увеличим число часов, если оно стало больше или равно 24, то перейдем на следующие сутки.

Пример решения на Python:

```
h, m = map(int, input().split())
for i in range(10):
    print(h, m)
    m += 9
    if m >= 60:
        h += 1
        m -= 60
    if h >= 24:
        h -= 24
```

## Р7. Акция

Ограничение времени 1 секунда

Ограничение памяти 64Mb

В книжном магазине проходит акция «3 по цене 2». Согласно условиям акции, каждый покупатель, который берет 3 книги получает самую дешевую из них бесплатно. Конечно, покупатель может взять больше книг и, в зависимости от того, как книги распределены в группы по три, получить самые дешевые из каждой группы бесплатно.



Например, пусть покупатель хочет взять книги со стоимостями: 10 3 2 4 6 4 9. Если разбить их на группы: (10, 3, 2), (4, 6, 4) и (9), он получит бесплатно книгу стоимостью 2 из первой группы и книгу стоимостью 4 из второй группы. Из третьей группы бесплатно получить ничего не получится, потому что она содержит только одну книгу.

Продавец книжного магазина очень дружелюбен к покупателям, и он всегда хочет снизить цену как можно больше. Для заданных цен книг, помогите ему распределить книги в группы наилучшим образом, так чтобы общая стоимость была минимальной.

Обратите внимание: не обязательно распределять книги в группы так, чтобы каждая группа содержала ровно 3 книги, но количество книг в каждой группе должно быть от 1 до 3, включительно.

### Формат ввода

Первая строка входного файла содержит целое число  $n$  ( $1 \leq n \leq 100$ ), число книг, которое покупатель хочет взять.

Каждая из следующих  $n$  строк содержит одно целое число  $c_i$  ( $1 \leq c_i \leq 1000$ ), цена каждой книги.

## Формат вывода

Выведите минимальную стоимость.

### Пример

Ввод	Вывод
5 6 5 9 2 8	24

### Решение.

Посмотрим, какую самую дорогую книгу мы можем получить бесплатно по акции. Это третья по цене книга. Для того, чтобы получить ее бесплатно, нужно купить группу из трех самых дорогих книг. Продолжая рассуждения тем же способом, получим, что далее, чтобы получить наибольшую скидку, нужно купить три следующие по цене книги, и т. д. Таким образом, получаем следующий алгоритм.

Отсортируем книги по убыванию цены. Формируем группы (1, 2, 3), (4, 5, 6), (7, 8, 9), ... пока книг хотя бы три. В каждой группе третья книга бесплатная.

Пример решения на Python:

```
n = int(input())
a = [int(input()) for i in range(n)]
a.sort()
a = a[::-1]
s = 0
for i in range(n):
    if i % 3 != 2:
        s += a[i]
print(s)
```

## Р8. Кенгуренок

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Кенгуренок хочет попасть из точки 0 в точку D. Он умеет прыгать только на 3 вперед или на 2 назад. То есть, из точки x он может попасть либо в точку x+3, либо в точку x-2. Помогите ему попасть в точку D за минимальное число прыжков.

## Формат ввода

Ввод содержит одно число  $D$  ( $1 \leq D \leq 100$ ).

## Формат вывода

В первой строке выведите число прыжков, которые нужно сделать кенгуренку. Во второй строке выведите сами прыжки.

### Пример 1

Ввод	Вывод
	3
9	3 3
	3

### Пример 2

Ввод	Вывод
	4
2	3 -2
	3 -2

## Решение.

Для того, чтобы число прыжков было минимальным, нужно делать как можно меньше прыжков назад. Если число  $d$  делится на 3, то можно не делать ни одного прыжка назад. Если остаток от деления на 3 равен 1, то нужно сделать один прыжок назад, если остаток равен 2, то нужно два прыжка назад. Чтобы упростить код, сделаем цикл **while**, который делает прыжки назад, пока расстояние до нужной точки не будет делиться на 3.

Пример решения на Python:

```
d = int(input())
ans = []
while d % 3 != 0:
    ans.append(-2)
    d += 2
while d > 0:
    ans.append(3)
    d -= 3
print(len(ans))
print(*ans)
```

Продолжение следует...