

Всероссийская олимпиада школьников по информатике, 2014-15 уч. год  
Первый (школьный) этап, г. Москва  
Разбор заданий для 7-8 классов

Каждая задача оценивается в 10 баллов. Итоговый балл выставляется как сумма баллов за 4 задачи с лучшим результатом (то есть для получения максимального балла нужно решить 4 любые задачи).

## Задача 1. Семизначное число

Придумайте натуральное число, которое удовлетворяет следующим условиям:

1. Запись числа состоит из семи цифр.
2. Сумма всех цифр числа равна 39.
3. В записи числа есть хотя бы одна цифра 4.
4. В записи числа есть хотя бы одна цифра 7.
5. Запись числа является палиндромом, то есть одинаково читается как слева направо, так и справа налево (например, такими числами-палиндромами являются числа 121 и 7007, но не является число 123).
6. Число является максимальным из всех чисел, удовлетворяющих пунктам 1-5.

В ответе запишите придуманное вами число.

### Решение

Семизначное число-палиндром состоит из трех цифр в начале числа, одной цифры посередине числа, и трёх цифр в конце числа, которые совпадают с первыми тремя цифрами.

Сумма всех цифр числа равна 39, и поскольку число является семизначным числом-палиндромом, то посередине десятичной записи числа стоит нечетная цифра. Значит, цифра 4 не стоит в числе посередине, и в числе содержится минимум две цифры 4 (по одной цифре среди первых трёх и среди последних трёх цифр числа).

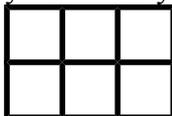
На первое место в числе поставим цифру 9, как максимальную цифру. Поскольку  $9 + 4 + 7 = 20$ , то число 7 не может встречаться среди первых трех цифр числа (сумма первых трех цифр числа не может быть больше 19), значит цифра 7 стоит посередине числа. Тогда сумма первых трех цифр числа равна  $(39 - 7) / 2 = 16$ , и поскольку две цифры из первых трех уже известны (это 9 и 4), то третья цифра из первых трех равна 3.

Поэтому максимальное число, удовлетворяющее условиям, равно 9437349.

## Задача 2. Прямоугольник в тетрадке

Скучая на уроке, Ваня нарисовал в тетрадке в клеточку прямоугольник со сторонами  $n \times m$  клеток. После чего он провел внутри прямоугольника все горизонтальные и вертикальные линии вдоль всех клеточек.

Например, если  $n = 2$ ,  $m = 3$ , то у Вани получилась следующая картинка.



Определите суммарную длину всех проведенных Ваней линий считая, что сторона одной клеточки равна единице.

Ответом к этой задаче является некоторое выражение, которое может содержать целые числа, переменные  $n$  и  $m$ , операции сложения (обозначается «+»), вычитания (обозначается «-»), умножения (обозначается «\*») и круглые скобки для изменения порядка

действий. Запись вида « $2n$ » для обозначения произведения числа 2 и переменной  $n$  неверная, нужно писать « $2 * n$ ».

Пример правильного (по форме записи) выражения:  $n + (m - 1) * 2$

## Решение

Вася нарисовал  $n + 1$  горизонтальную линию длины  $m$  и  $m + 1$  вертикальную линию длины  $n$ .

Ответ:  $(n + 1) * m + (m + 1) * n$ .

## Задача 3. Лексикографический порядок

Все пятибуквенные слова, составленные из латинских букв F, N, S записаны в алфавитном порядке. Вот начало этого списка:

1. FFFFF
2. FFFFN
3. FFFFS
4. FFFNF
5. ...

Определите слова, которые будут идти в этом списке под номерами 7, 27, 90, 101, 196. Возможно, вам будет проще ответить на третий и четвертый вопрос если вы будете знать, что на 99-м месте в этом списке стоит строка NFNS.

В ответе нужно записать пять строк, состоящих из латинских букв. Ответ на каждое задание нужно писать в отдельной строке (в первой строке — слово, стоящее в списке 7-м, во второй строке — слово, стоящее 27-м, в третьей строке — 90-м, в четвертой строке — 101-м, в пятой строке — 196-м). Порядок записи слов в ответе менять нельзя. В ответе должно быть ровно пять строк. Если вы не можете найти какой-то из ответов, вместо него напишите любую строчку из данных пяти букв.

## Решение

Такие последовательности связаны с троичной системой счисления. Если заменить букву F на цифру 0, букву N на цифру 1, а букву S на цифру 2, то получится такой список:

1. 00000
2. 00001
3. 00002
4. 00010

То есть в строке номер  $i$  записано представление в троичной системе счисления числа  $i - 1$ , дополненного нулями до пяти цифр.

Число, стоящее в строке 7, можно получить из числа, стоящего в строке 4, прибавлением 3, то есть при помощи увеличения на 1 второй справа цифры. Получим последовательность FFFSF.

Число, стоящее в строке 27, является троичной записью числа  $26 = 3^3 - 1$ . Число  $3^3 - 1$  в троичной системе счисления записывается, как 222, поэтому ответом на второй вопрос будет FFSS.

Для ответа на третий вопрос (что записано в строке 90) можно использовать информацию о том, что в строке 99 записано NFNS. Для получения ответа нужно уменьшить число на 9, то есть уменьшить на 1 третью справа цифру числа. Ответом будет строка NFFSS.

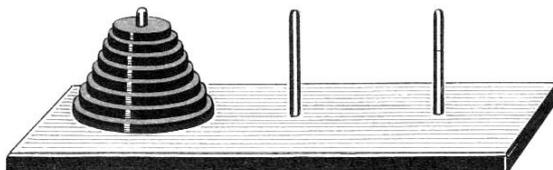
Для ответа на четвертый вопрос строку NFNS нужно увеличить на 2. После увеличения на 1 получится NFSFF, после повторного увеличения получится NFSFN.

Наконец, для ответа на последний вопрос (что записано в строке 196) переведем число 195 в троичную систему счисления. Получится число 21020, поэтому ответом на последний вопрос будет строка SNFSF.

Ответ:  
FFFSF  
FFSSS  
NFFSS  
NFSFN  
SNFSF

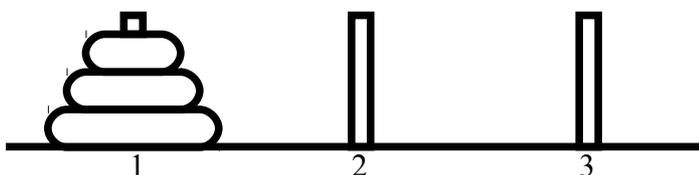
## Задача 4. Сломанная Ханойская башня

Известная головоломка «Ханойская башня» представляет собой три стержня, на один из которых надета пирамидка из нескольких дисков разного размера. Например, головоломка с восемью дисками может выглядеть вот так:



Необходимо переложить пирамидку с одного из трёх стержней на другой. При этом за один ход можно переложить только один диск с одного стержня (взяв самый верхний диск, лежащий на этом стержне) на другой стержень, при этом нельзя класть диск большего размера на меньший по размеру диск. Например, если взять верхний диск с первого стержня и переложить на второй стержень, то следующим ходом можно переложить верхний диск с первого стержня на третий стержень, так как на втором стержне будет лежать меньший диск.

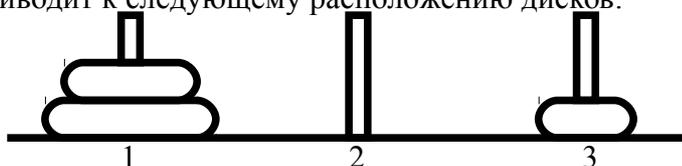
В этой задаче будет рассматриваться головоломка из трех дисков. **При этом в этой задаче запрещено перекладывать диски со стержня номер 1 на стержень номер 3 и, наоборот, со стержня номер 3 на стержень номер 1.** Во всем остальном правила оригинальной головоломки сохраняются.



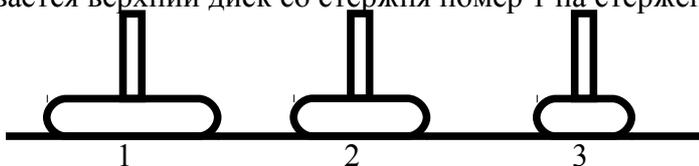
Вам необходимо придумать алгоритм перекладывания всей пирамидки из трех дисков со стержня номер 1 на стержень номер 2. Алгоритм записывается следующим образом. Каждая строчка содержит одну команду перемещения. В строке записывается два числа через пробел, номер стержня, с которого берется верхний диск и номер стержня, на который перекладывается взятый диск. Например, рассмотрим следующий алгоритм перекладывания:

1 3
1 2
3 2

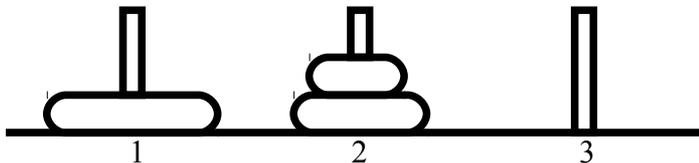
В этом алгоритме сначала перекладывается верхний диск со стержня номер 1 на стержень номер 3, что приводит к следующему расположению дисков:



Затем перекладывается верхний диск со стержня номер 1 на стержень номер 2:



Наконец, перекладывается диск со стержня номер 3, на стержень номер 2:



Этот алгоритм приведен только для примера записи, и не может быть частью верного решения этой задачи, поскольку в этой задачи, как было сказано ранее, запрещено перекладывать диски между стержнями 1 и 3 и наоборот.

В ответе запишите алгоритм перекладывания пирамидки из трех дисков со стержня номер 1 на стержень номер 2, не содержащий запрещенных перекладываний. Чем меньше перекладываний будет в вашем алгоритме, тем больше баллов вы получите.

## **Решение**

Головоломку можно решить за 13 перекладываний:

1 2  
2 3  
1 2  
3 2  
2 1  
2 3  
1 2  
2 3  
1 2  
3 2  
2 1  
3 2  
1 2

Решением задач 5–7 является программа, написанная на одном из языков программирования. Задачи 5–7 необязательно решать для получения полного балла.

Ограничение по времени работы программы в задачах 5–7: 1 секунда.

Задачи принимаются на проверку и оцениваются только если они выдают правильный ответ на примере входных и выходных данных, приведенном в условии задачи. Программа не должна выводить никаких иных сообщений, кроме того, что требуется в задаче.

## Задача 5. Шахматная доска

Шахматная доска состоит из  $n \times m$  клеток, покрашенных в черный и белый цвет в «шахматном» порядке. При этом клетка в левом нижнем углу доски покрашена в черный цвет. Определите, сколько всего на доске черных клеток.

Программа получает на вход два числа  $n$  и  $m$ , записанных в отдельных строках. Все числа — натуральные, не превосходящие 30 000.

Программа должна вывести одно целое число — количество черных клеток на доске.

### Пример входных и выходных данных

Ввод	Вывод
3	6
4	

### Система оценивания

Решение, правильно работающее только для случаев, когда входные числа не превосходят 10, будет оцениваться в 4 балла.

### Решение

Если на доске — четное число клеток, то черных и белых клеток поровну, поэтому ответом будет  $nm/2$ . Если же произведение  $nm$  нечетно, то черных клеток будет на одну больше, поэтому можно поделить  $nm$  на 2 нацело и прибавить к результату 1.

Решения задач будут приведены на языке Python версии 3. В языке Python для целочисленного деления используется операция «//» (аналог операции div в языке Паскаль, функции div в алгоритмическом языке или операции «\» в Бейсике). Для проверки четности используется в языке Python операция «%» (аналог операции mod в языках Паскаль и Бейсик и функции mod в алгоритмическом языке).

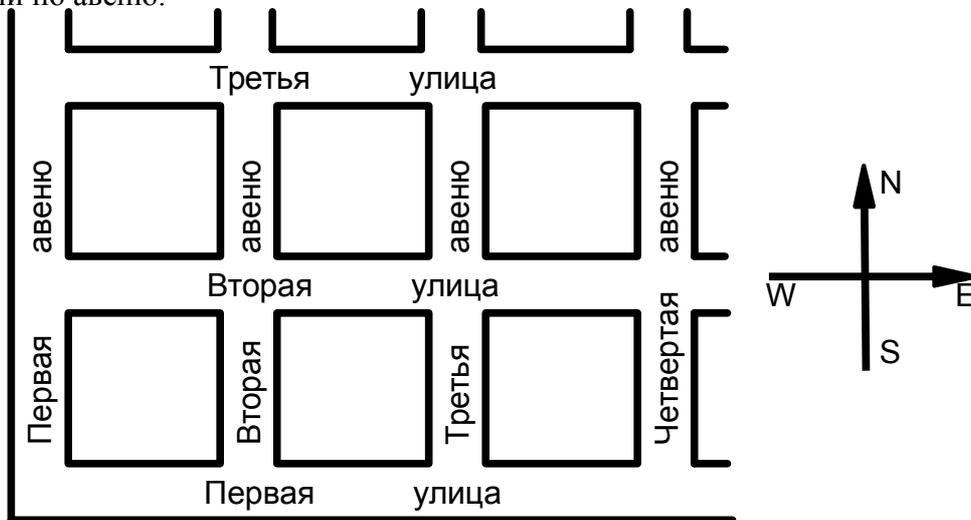
```
n = int(input())
m = int(input())
if n * m % 2 == 0:
    print(n * m // 2)
else:
    print(n * m // 2 + 1)
```

Можно решить эту задачу и без использования инструкции if, если заметить, что результатом является частное от деления  $nm$  на 2, округленное вверх:

```
n = int(input())
m = int(input())
print((n * m + 1) // 2)
```

## Задача 6. Манхэттен

Кварталы Манхэттена состоят из авеню, направленных с юга на север и улиц, направленных с запада на восток. Все улицы и авеню пронумерованы числами, начиная с 1 подряд (первая улица, вторая улица, третья улица и т. д.). Передвигаться можно только по улицам или по авеню.



Миша впервые попал на Манхэттен. Сейчас он стоит на пересечении авеню номер  $x_1$  и улицы номер  $y_1$ . Ему нужно попасть на перекресток авеню номер  $x_2$  и улицы номер  $y_2$ . Определите маршрут, который он должен пройти.

Программа получает на вход 4 числа:  $x_1$ ,  $y_1$ ,  $x_2$ ,  $y_2$ , записанных в отдельных строках. Все числа — натуральные, не превышают  $10^3$ . Начальное и конечное расположение Миши не совпадают.

Программа должна вывести последовательность из латинских заглавных букв, описывающих маршрут, которому должен следовать Миша. Буква «N» обозначает перемещение на один квартал на север, «S» — на юг, «W» — на запад, «E» — на восток. Программа должна вывести самый короткий из всех возможных маршрутов, если же кратчайших маршрутов существует несколько, то программа должна вывести любой из них (но только один).

Программа может выводить последовательность ходов не в одну строку (как в примере), а каждый символ ответа — в отдельной строке (например, если каждый символ ответа выводится при помощи отдельной команды вывода внутри цикла).

### Пример входных и выходных данных

Ввод	Вывод	Примечание
1 3 4 1	EEEESS	Миша стоит на пересечении первой авеню и третьей улицы и должен попасть на пересечение четвертой авеню и первой улицы. Ему нужно пройти три квартала на восток и два квартала на юг. Возможны и другие ответы, например, «SSEEE», «ESESE» и прочие. Возможен вывод ответа не в одну строку, а каждый символ ответа — в отдельной строке

### Решение

Если  $x_2 > x_1$ , то Миша должен переместиться на  $x_2 - x_1$  кварталов на восток, поэтому программа должна вывести  $x_2 - x_1$  символ «E», что можно сделать при помощи цикла. Иначе

Миша должен переместиться на  $x_1 - x_2$  квартала на запад, поэтому программа должна вывести  $x_1 - x_2$  символ «W». Аналогично рассматривается передвижение на север и на юг. Пример решения на языке Python, использующие циклы для вывода последовательности перемещений:

```
x1 = int(input())
y1 = int(input())
x2 = int(input())
y2 = int(input())
if x2 > x1:
    for i in range(x2 - x1):
        print("E")
else:
    for i in range(x1 - x2):
        print("W")
if y2 > y1:
    for i in range(y2 - y1):
        print("N")
else:
    for i in range(y1 - y2):
        print("S")
```

Приведем еще один пример решения, в котором для повторения одного символа используется операция умножения строки на число, то есть для того, чтобы повторить символ «E»  $x_2 - x_1$  раз нужно выполнить операцию "E"\* ( $x_2 - x_1$ ). При этом если умножить строку на отрицательное число, то получится пустая строка:

```
x1 = int(input())
y1 = int(input())
x2 = int(input())
y2 = int(input())
print("E"*(x2-x1) + "W"*(x1-x2) + "N"*(y2-y1) + "S"*(y1-y2))
```

## Задача 7. Пятница, 13-е

Календарь жителей планеты Плюк состоит из  $N$  месяцев, каждый месяц состоит ровно из 30 дней, неделя состоит из 7 дней. Особо несчастливными на планете Плюк считается 13-е число месяца, если оно выпадает на пятницу.

Известно, что Новый год на планете Плюк начался в  $k$ -й по счету день недели (1-й день недели — понедельник, 2-й — вторник, 3-й — среда, ... , 7-й — воскресенье). Определите, сколько в этом году на планете Плюк будет особо несчастливых пятниц, 13-е.

Программа получает на вход два натуральных числа, записанных в отдельных строках. Первое число — количество месяцев в календаре планеты Плюк  $N$ , не превосходящее  $10^9$ . Второе число — номер дня недели, на который приходится первое число первого месяца нового года, может принимать значения от 1 до 7.

Программа должна вывести единственное целое число — количество несчастливых дней в этом году.

### Пример входных и выходных данных

Ввод	Вывод	Примечание
12 1	2	На 13-е число будут приходиться пятницы четвертого и одиннадцатого месяцев.

## Система оценивания

Решение, правильно работающее только для случаев, когда  $N$  не превосходит 100, будет оцениваться в 6 баллов.

## Решение

Решение на 6 баллов использует простое моделирование. Сначала посчитаем, на какой день недели приходится 13-е число первого месяца, затем будем считать, на какой день недели приходится 13-е число каждого следующего месяца. Если пронумеровать дни недели от 1 до 6, а воскресенье считать днём недели номер 0, то для получения дня недели 13-го числа следующего месяца нужно к текущему номеру дня недели прибавить 30 и взять остаток от деления на 7. Повторяем это в цикле  $N$  раз и при получении числа 5 увеличиваем ответ на 1. Пример решения на языке Python:

```
N = int(input())
k = int(input())
ans = 0
k = (k + 12) % 7
for i in range(N):
    if k == 5:
        ans += 1
    k = (k + 30) % 7
print(ans)
```

Такое решение будет получать результат «превышено максимальное время работы» при больших значениях  $N$ , так как за 1 секунду невозможно выполнить цикл в  $10^9$  шагов.

Для решения задачи на полный балл заметим, что поскольку числа 7 и 30 — взаимно простые, то в последовательности дней недели, на которые выпадают 13-е числа месяца будет цикл длины 7: пятница, воскресенье, вторник, четверг, суббота, понедельник, среда. Поэтому для нахождения ответа достаточно найти первый месяц года, в котором будет пятница 13-е, и к ответу добавить количество оставшихся месяцев в году, деленное на 7 нацело. Пример решения, набирающего полный балл:

```
N = int(input())
k = int(input())
k = (k + 12) % 7
i = 1
ans = 0
while i <= N and k != 5:
    i += 1
    k = (k + 30) % 7
if i > N:
    print(0)
else:
    print(1 + (N - i) // 7)
```