

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача А: Сложная арифметика**

Мудрый филин Гуамоку учит Страшилу арифметике и хочет придумать для него как можно больше примеров на деление.

Филин берет два каких-нибудь натуральных числа  $A$  и  $B$  и проверяет, делится ли  $A$  на  $B$  без остатка. Если  $A$  не делится на  $B$ , филин пытается изменить числа так, чтобы в итоге  $A$  разделилось на  $B$ . Так как он и сам не очень силен в арифметике, то изменять числа филин может только увеличивая или уменьшая их на 1.

Операция уменьшения или увеличения числа на 1 занимает ровно 1 секунду. Например, чтобы превратить число 5 в число 2, филину потребуется 3 секунды, так как ему придется три раза выполнить операцию уменьшения числа на 1. Аналогично, превращение числа 8 в число 10 займет 2 секунды.

Напишите программу, которая по двум заданным числам  $A$  и  $B$  определит, за какое минимальное время филин Гуамоку может так изменить числа  $A$  и  $B$ , что  $A$  будет делиться на  $B$  без остатка.

**Формат входных данных**

Единственная строка входных данных содержит два натуральных числа  $A$  и  $B$  ( $1 \leq A, B \leq 10^5$ ), разделенные пробелом.

**Формат выходных данных**

Выведите единственное число - минимальное время, за которое филин может изменить числа  $A$  и  $B$  так, что  $A$  будет делиться на  $B$  без остатка.

**Пример входных данных**

35 19

**Пример выходных данных**

2

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача В: Сумма цифр произведения**

Обозначим число, которое в десятичной системе счисления состоит только из  $k$  единиц, как  $R(k)$ . Например,  $R(3)=111$ ,  $R(5)=11111$ .

Для заданного количества цифр  $n$  и числа  $A$ , вычислите сумму цифр числа  $R(n)*A$ .

**Формат входных данных**

В первой строке записано число  $T$  ( $1 \leq T \leq 10^5$ ) - количество наборов входных данных.

В каждой из последующих  $T$  строк записаны через пробел два числа  $n$ ,  $A$  ( $1 \leq n \leq 10^{16}$ ,  $1 \leq A \leq 10^6$ ) - длина первого множителя и второй множитель.

**Формат выходных данных**

Для каждой пары чисел из входного набора данных выведите в отдельной строке единственное число - сумму цифр в соответствующем произведении.

**Пример входных данных**

```
2
3 4
4 12
```

**Пример выходных данных**

```
12
12
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача С: Треугольный пирог**

Торт в форме прямоугольного треугольника лежит на плоскости. Две стороны этого треугольника параллельны осям координат.

Друзья хотят поделить его на  $n$  равных по площади частей, сделав  $n-1$  вертикальный разрез. Помогите найти координаты  $x$  точек на стороне треугольника, через которые эти разрезы должны пройти.

**Формат входных данных**

В первой строке записано число  $T$  ( $1 \leq T \leq 3000$ ) - количество наборов входных данных. Далее следуют описания наборов.

Каждый набор входных данных состоит из четырех строк. Первая строка содержит число  $n$  ( $2 \leq n \leq 100$ ) - количество частей, на которые нужно разделить торт.

В следующих 3 строках записаны по два целых числа, разделённых пробелом,  $x_i, y_i$  ( $0 \leq x_i, y_i \leq 10^4$ ) - координаты  $i$ -го угла треугольника.

Гарантируется, что треугольник прямоугольный, имеет ненулевую площадь и две его стороны параллельны осям координат.

**Формат выходных данных**

Для каждого набора входных данных выведите строку, состоящую из  $n-1$  вещественного числа - координат  $x$  точек на стороне треугольника, через которые нужно провести вертикальные разрезы. Координаты должны быть выведены через пробел в порядке возрастания с точностью не менее 6 знаков после запятой.

**Пример входных данных**

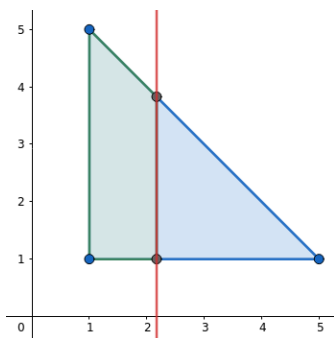
```
2
2
1 1
1 5
5 1
3
1 5
1 1
5 5
```

**Пример выходных данных**

```
2.1715729
1.7340137 2.6905989
```

**Замечания**

Иллюстрация к первому тесту. Красная линия делит треугольник на части, каждая из которых имеет площадь 4.



**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача D: Красивая шеренга - 2**

Как вы, наверное, знаете, Урфин Джюс любит, что бы его дуболомы образовывали красивый строй. Недавно он выяснил, что шеренга дуболомов будет красивой, если дуболомы стоят в ней по росту - первый дуболом должен быть самым высоким, а каждый следующий дуболом должен быть не выше предыдущего.

Урфин велел дуболомам построиться, и они, конечно же, выстроились в шеренгу как попало. Для того, чтобы сделать эту шеренгу красивой, Урфин решил поступить просто - он убрал некоторое количество подряд стоящих дуболомов из начала шеренги, и некоторое количество подряд стоящих дуболомов из конца шеренги так, чтобы оставшиеся дуболомы были упорядочены по росту от самого высокого к самому низкому.

Вам будет дан массив, содержащий рост дуболомов в первоначальной шеренге, начиная с первого дуболома. Определите длину самой длинной красивой шеренги, которую Урфин Джюс может получить описанным выше способом.

**Формат входных данных**

Первая строка входных данных содержит единственное число  $N$  ( $1 \leq N \leq 10^5$ ) - первоначальное количество дуболомов в шеренге.

Следующая строка содержит  $N$  натуральных чисел  $H_i$  ( $1 \leq H_i \leq 1000$ ) - рост дуболомов, начиная с самого первого.

**Формат выходных данных**

Выведите единственное число - длину самой длинной красивой шеренги, которую может получить Урфин Джюс описанным в условии способом.

**Пример входных данных**

```
8
1 3 7 5 5 3 5 2
```

**Пример выходных данных**

```
4
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача Е: Ограниченные суммы**

Теперь мудрый филин Гуамоку придумывает для Страшила задачи на сложение. Филин берет некоторое число  $S$  и раскладывает его на слагаемые разными способами. Например, для числа 4 филин получил 8 разных способов (способы, в которых используются одни и те же слагаемые, но в другом порядке, считаются разными).

$4=1+1+1+1$ ;  $4=1+1+2$ ;  $4=1+2+1$ ;  $4=2+1+1$ ;  $4=2+2$ ;  $4=3+1$ ;  $4=1+3$ ;  $4=4$  (да, в Изумрудном городе бывают суммы из одного слагаемого).

Так как Страшила умеет суммировать числа не очень хорошо, Гуамоку решил оставить наборе сумм только такие суммы, в которых разность между максимальным и минимальным слагаемым не превышает некоторое заданное число  $K$ . Эти суммы и будут теми примерами на сложение, которые должен будет решить Страшила.

Но даже мудрый филин Гуамоку не может посчитать, сколько же примеров нужно будет решить Страшале.

Напишите программу, которая по заданным числам  $S$  и  $K$  посчитает количество способов представить число  $S$  в виде суммы нескольких слагаемых таким образом, чтобы разность между максимальным и минимальным слагаемым не превосходила  $K$ .

Способы, отличающиеся перестановкой слагаемых, считаются разными. Так как число способов может быть очень велико, в качестве ответа выведите остаток от деления полученного числа способов на число  $10^9+7$ .

**Формат входных данных**

В единственной строке входных данных записаны два натуральных числа  $S$  и  $K$  ( $1 \leq S, K \leq 100$ ), разделенные пробелом.

**Формат выходных данных**

Выведите единственное число - ответ на задачу.

**Пример входных данных**

4 1

**Пример выходных данных**

6

**Замечания**

**Пояснение к примеру**

Так как разность между максимальным и минимальным слагаемым не должна быть больше 1, то суммы  $4=3+1$  и  $4=1+3$  не подходят, и из приведенных в условии 8 сумм подходят оставшиеся 6.

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача F: Ценные изумруды**

Гудвин, великий и ужасный правитель изумрудного города, захотел систематизировать свою коллекцию изумрудов. Он решил разбить множество своих изумрудов на несколько непересекающихся подмножеств. Два изумруда должны попасть в одно подмножество, если их стоимости имеют общий делитель, больший 1. Ценностью подмножества Гудвин считает стоимость самого дорогого изумруда в этом подмножестве. Если существует несколько вариантов разбиения, Гудвин выберет тот, в котором количество подмножеств максимально.

Вам будет дана секретная информация про стоимости всех изумрудов Гудвина. Посчитайте произведение ценностей подмножеств, на которые Гудвин разобьет множество своих изумрудов по описанному выше правилу.

**Формат входных данных**

Первая строка входных данных содержит единственное целое число  $N$  ( $1 \leq N \leq 1000$ ) - количество изумрудов у Гудвина.

Вторая строка содержит  $N$  натуральных чисел  $A_i$ , разделенных пробелами ( $1 \leq A_i \leq 10^9$ ) - стоимости изумрудов.

**Формат выходных данных**

Найдите единственное число - произведение ценностей всех подмножеств, на которые Гудвин разобьет множество изумрудов по указанному в условии правилу. Так как это число может быть очень большим, выведите остаток от деления найденного числа на  $10^9+7$ .

**Пример входных данных**

```
4
2 4 3 9
```

**Пример выходных данных**

```
36
```

**Пример входных данных**

```
3
2 3 6
```

**Пример выходных данных**

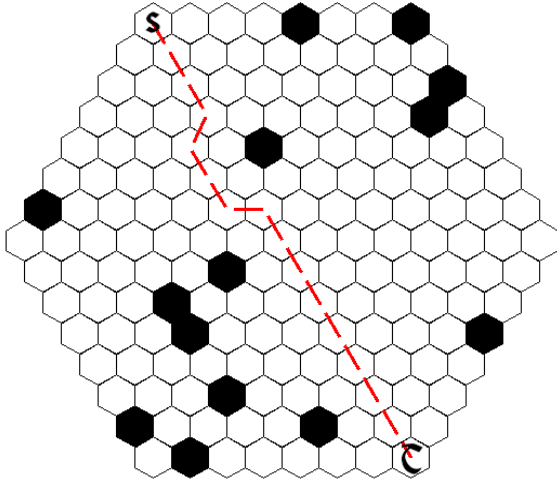
```
6
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача G: Побег**

Страшила сбежал из плена Урфина Джюса и теперь хочет встретиться со своими друзьями. У него есть фрагмент карты Волшебной страны, на котором страна изображена в виде большого правильного шестиугольника, состоящего из более мелких шестиугольных клеток. Каждая сторона карты состоит из  $n$  клеток. На карте отмечены все отряды дуболомов. От вороны Кагги-Карр Страшила знает, где сейчас находится Трусливый Лев, и хотел бы добраться к нему самым лучшим способом.



Путём назовём последовательность клеток, в которой соседние клетки имеют общую сторону. Путь не может выходить за пределы заданного фрагмента. Длина пути это количество переходов между клетками в пути. Расстояние между двумя клетками это длина кратчайшего пути между ними.

Определим надёжность пути как минимальное расстояние от какой-то клетки пути до клетки с отрядом дуболомов. Страшила считает путь оптимальным, если его надёжность максимально возможная. Помогите ему найти надёжность оптимального пути, который начинается в клетке, где он сейчас находится, и заканчивается в клетке, где находится Трусливый Лев.

**Формат входных данных**

В первой строке содержится единственное число  $n$  ( $2 \leq n \leq 100$ ) - количество шестиугольников вдоль каждой из сторон карты.

В следующих  $2n-1$  строках содержится описание фрагмента карты.  $i$ -я строка начинается с некоторого числа пробелов, после которых идут разделённые пробелами символы, описывающий содержимое клетки:

- . - пустая клетка
- D - отряд дуболомов
- S - Страшила
- T - Трусливый Лев

Гарантируется, что клетки образуют правильный шестиугольник со стороной  $n$  клеток, есть хотя бы одна занятая клетка, а также ровно одна начальная и конечная клетки. На карте всегда есть путь, из стартовой в конечную клетку, который не проходит через клетки с дуболомами.

**Формат выходных данных**

Выведите строку с единственным числом - надёжность оптимального пути, который начинается в клетке, где сейчас находится Страшила и заканчивается в клетке, где находится Трусливый Лев.

**Пример входных данных**

```
3
. . . S
. . . .
. . D . .
. . . .
T . .
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Пример выходных данных**

2

**Пример входных данных**

8

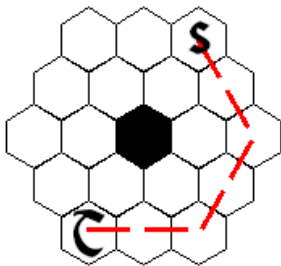
```
      S . . . . D . . . D
      . . . . . . . . . .
      . . . . . . . . . . D
      . . . . . . . . . . D .
      . . . . . D . . . . .
      . . . . . . . . . . .
      D . . . . . . . . . .
      . . . . . . . . . . .
      . . . . . D . . . . .
      . . . . . D . . . . .
      . . . . . D . . . . . D
      . . . . . . . . . . .
      . . . . . D . . . . .
      D . . . . . D . . . .
      . D . . . . . . . T
```

**Пример выходных данных**

2

**Замечания**

Первому тесту соответствует следующая иллюстрация:



Второму тесту соответствует иллюстрация из условия.



**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача Н: Машинное обучение**

По совету умного филина Гуамоку Урфин Джнос для систематизации своей коллекции изумрудов решил применить современный метод машинного обучения - метод  $k$ -ближайших соседей.

Зададим числовую величину, которая будет задавать похожесть двух изумрудов. Будем называть эту величину расстоянием между изумрудами. Для простоты будем считать, что изумруд - это шар радиуса  $R$  и цвета  $C$  (градация зеленого цвета). Урфин решил вычислять расстояние между изумрудами, то есть степень их похожести, по формуле  $d = \max(|R_1 - R_2|, |C_1 - C_2|)$ , где целые положительные числа  $R_1$  и  $R_2$  - это радиусы изумрудов, а целые положительные числа  $C_1$  и  $C_2$  - их цвета. Чем меньше величина  $d$ , тем ближе изумруды друг к другу, то есть тем более эти два изумруда похожи.

Первоначально Урфин выбрал  $N$  самых красивых с его точки зрения изумрудов. Каждый из этих изумрудов является единственным представителем своего класса. Теперь необходимо каждый из оставшихся  $M$  изумрудов отнести к какому-нибудь из этих  $N$  первоначальных классов. Для этого Урфину необходимо выполнить следующие шаги.

1. Для текущего изумруда найти  $k$  самых близких к нему изумрудов (то есть самых похожих на него).
2. Выяснить, изумрудов какого класса больше всего среди этих  $k$  найденных ближайших соседей.
3. Номер этого класса и присваиваем текущему изумруду.

Другими словами текущий изумруд получает номер того класса, которому принадлежит большинство из его  $k$ -ближайших соседей. После того, как очередной изумруд классифицирован, он принимает участие в классификации оставшихся изумрудов.

Напишите программу, которая поможет Урфину определить описанным выше методом номер класса для каждого из  $M$  неклассифицированных изумрудов.

Если на некотором шаге количество классифицированных изумрудов меньше  $k$ , то при поиске ближайших соседей рассматриваем все изумруды. Если невозможно выбрать  $k$  ближайших изумрудов, выбираем минимальное число, большее  $k$ . Если сразу несколько классов содержат наибольшее количество выбранных изумрудов, выбираем класс с наименьшим номером.

**Формат входных данных**

Первая строка входных данных содержит три натуральных числа -  $N, k, M$  ( $1 \leq N, k, M \leq 1000$ ).

В каждой из последующих  $N$  строк содержатся два числа, разделенные пробелом,  $R_i, C_i$  ( $1 \leq R_i, C_i \leq 1000$ ) - радиус и цвет изумруда, представляющего  $i$ -ый класс.

В каждой из последующих  $M$  строк содержатся два числа, разделенные пробелом,  $R_j, C_j$  ( $1 \leq R_j, C_j \leq 1000$ ) - радиус и цвет изумрудов, которые необходимо отнести к одному из  $N$  классов.

**Формат выходных данных**

Выведите ровно  $M$  строк. В  $j$ -ой строке выведите единственное число - номер класса, которому принадлежит  $j$ -ый изумруд из  $M$  изумрудов, которые надо классифицировать. Классы нумеруются, начиная с 1.

**Пример входных данных**

```
3 1 4
3 1
8 2
4 7
7 3
2 1
7 4
3 6
```

**Пример выходных данных**

```
2
1
2
3
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Пример входных данных**

```
3 2 4
4 6
4 4
4 2
6 4
5 5
5 4
6 6
```

**Пример выходных данных**

```
1
1
1
1
```

**III Командная олимпиада школьников Крыма по программированию**  
**Финальный тур**  
**21 мая 2022 г.**

---

**Задача I: Рекурсивный узор**

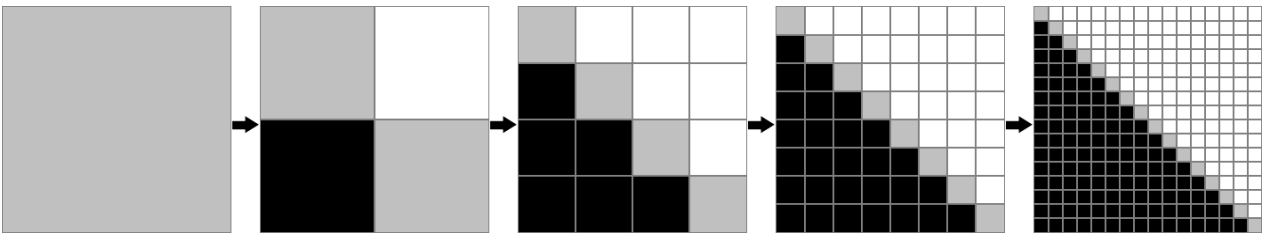
Возьмем на листе в клеточку квадрат размером  $n \times n$  клеток и закрасим некоторые из них в белый, серый или чёрный цвет. Полученный квадрат представляет собой шаблон будущего узора.

Для построения узора воспользуемся следующим алгоритмом:

1. Берем чистый квадратный лист серого цвета
2. ШАГ 1. Лист делится на  $n^2$  одинаковых квадратов. Эти квадраты раскрашиваются в соответствии с шаблоном: квадраты, соответствующие белым клеткам шаблона закрашиваются белым цветом, а чёрные - чёрным. Серые клетки остаются не закрашенными.
3. ШАГ 2. Каждый из квадратов, оставшихся серым, разбивается на  $n^2$  меньших квадратов, некоторые из них закрашиваются в соответствии с шаблоном.
4. На каждом из последующих шагов повторяются действия ШАГА 2.

После  $k$  шагов весь лист можно разбить на  $n^k \times n^k$  квадратных клеток, каждая клетка задаётся номером строки и номером столбца. Левый верхний угол всего листа имеет координаты  $(1, 1)$ , а правый нижний -  $(n^k, n^k)$ .

Напишите программу, которая по заданному  $n$  и количеству шагов  $k$  рисует фрагмент рекурсивного узора, попавший в заданный прямоугольник.



**Формат входных данных**

В первой строке записано число  $n$  ( $2 \leq n \leq 4$ ) - размер шаблона. В следующих  $n$  строках записано ровно  $n$  символов -- шаблон для построения узора. Шаблон состоит из символов 'W', '#' и 'B'. Эти символы задают клетки белого, серого, чёрного цвета соответственно.

В следующей строке записано число  $k$  ( $1 \leq k \leq 28$ ) - количество шагов алгоритма, которые нужно выполнить, чтобы получить искомый узор.

В последней строке записаны 4 числа  $r_1, c_1, r_2, c_2$  ( $1 \leq r_1 \leq r_2 \leq n^k, 1 \leq c_1 \leq c_2 \leq n^k, (r_2-r_1+1)(c_2-c_1+1) \leq 300000$ ) строка и столбец левого верхнего и правого нижнего угла прямоугольника, узор в котором нужно получить.

**Формат выходных данных**

Выведите матрицу размера  $r_2-r_1+1 \times r_2-r_1+1$  из символов 'W', '#' и 'B' - искомый фрагмент.

**Пример входных данных**

```
2
#W
B#
2
1 1 4 4
```

**Пример выходных данных**

```
#WWW
B#WW
BB#W
BBB#
```

**Пример входных данных**

```
2
#W
B#
```

