

# 1. ПЕРВЫЙ ЭТАП

# Задачи первого этапа. Математика.

## 1.1. Первая попытка. Задачи 9 класса.

### *Задача 1.1.1. (20 баллов)*

Гоша взял у друга 11 гаек М6 (ГОСТ 5916-70) и положил в карман рюкзака. Согласно ГОСТу 1 гайка М6 весит 1.254 грамма. И вот незадача, придя домой, Гоша насчитал в кармане 12 внешне одинаковых гаек! Одна из них была из того набора, что когда-то был куплен на блошином рынке, и, по его личному опыту, такие гайки имеют меньший вес, около грамма, а также сами по себе более низкого качества менее прочные.

У Гошиного папы есть весы, состоящие из двух больших чаш на двух концах рычага. За какое минимальное количество взвешиваний можно найти ту самую низкокачественную гайку?

### *Решение*

Если мы имеем  $n$  внешне одинаковых объектов, после одного взвешивания останется в худшем случае  $\lceil n/2 \rceil$  объектов, среди которых есть отличающийся по весу. Таким образом, если  $n = 12$ , то, в худшем случае, понадобятся 3 взвешивания  $12 \Rightarrow 4 \Rightarrow 2 \Rightarrow 1$ .

**Ответ:** 3.

### *Задача 1.1.2. (20 баллов)*

Витя приклеивал цифры номера квартиры на дверь. Этот номер состоит из трех цифр. В процессе приклеивания Вите пришла необычная мысль, что если в номере квартиры поменять местами две последние цифры и сложить получившееся число с исходным, то получится номер его школы! Юноша учился в школе 1187. Найдите все такие номера квартир, и если Витя живет в квартире с наименьшим из них, то в какой квартире он живет?

### *Решение*

Пусть номер квартиры равен  $\overline{abc} = 100 \cdot a + 10 \cdot b + c$ , где  $a$ ,  $b$  и  $c$  – цифры числа. Число с перевернутыми двумя последними цифрами при этом будет равно

$\overline{acb} = 100 \cdot a + 10 \cdot c + b$ . Затем решим уравнение в целых числах

$$100 \cdot a + 10 \cdot b + c + 100 \cdot a + 10 \cdot c + b = 1187,$$

$$200 \cdot a + 11 \cdot b + 11 \cdot c = 1187$$

с учетом ограничений на то, что  $a$ ,  $b$  и  $c$  – цифры в десятичной системе счисления.

$$200 \cdot a + 11 \cdot (b + c) = 200 \cdot 5 + 11 \cdot 17.$$

Таким образом, возможные номера квартир 589 и 598. Выбираем наименьший – 589.

**Ответ:** 589.

### **Задача 1.1.3. (20 баллов)**

Маша и Андрей, будущие математики, развлекались на перемене. Маша написала на доске 4 различных натуральных числа. Андрей выписал значения наибольших общих делителей для каждой из шести пар чисел. Получилось, что для одной из пар НОД равен 1, для другой – 2, для третьей – 3, для четвертой – 4, для пятой – 5, а для шестой –  $X$ . Найдите наименьшее возможное значение  $X$ ?

#### **Решение**

Пусть на доске записаны 4 числа  $a$ ,  $b$ ,  $c$  и  $d$ . Пусть  $\text{НОД}(a, b) = 2$ , тогда  $\text{НОД}(c, d) = 4$  быть не может, так как НОД всех пар будет кратным 2. Значит, пусть  $\text{НОД}(a, c) = 4$  и тогда  $d$  будет нечётным числом.

Исходя из записанных равенств, можно выписать следующие:

$$a = 4 \cdot a_4 = 2 \cdot a_2$$

$$b = 2 \cdot b_2$$

$$c = 4 \cdot c_4$$

При этом  $\text{НОД}(a_2, b_2) = 1$  и  $\text{НОД}(a_4, c_4) = 1$ . Очевидно также, что  $\text{НОД}(b, c) = 2 \cdot x$  – то есть это последняя искомая пара. Попробуем подобрать значения  $a$ ,  $b$ ,  $c$  и  $d$  так, чтобы они удовлетворяли всем равенствам и из НОД оставшихся пар равнялись указанным значениям. Например,  $a = 4$ ,  $b = 10$ ,  $c = 12$  и  $d = 15$ . Таким образом, наименьшее возможное значение равно 2.

**Ответ:** 2.

### **Задача 1.1.4. (20 баллов)**

Женя загадал некоторое число: его наименьший делитель (не равный 1) на 77 меньше наибольшего делителя (не равного самому числу). Чему равно это число? Укажите наименьшее из возможных.

**Решение**

Пусть загаданное число  $X = A \cdot B$ , где  $A$  – наименьший делитель,  $B$  – наибольший. Тогда  $A + 77 = B$ .  $X = A \cdot (A + 77) \rightarrow \min$ . Так как  $A \geq 1$ , то минимум достигается при  $A = 2$ . Таким образом,  $X = 2 \cdot 79 = 158$ .

**Ответ:** 158.

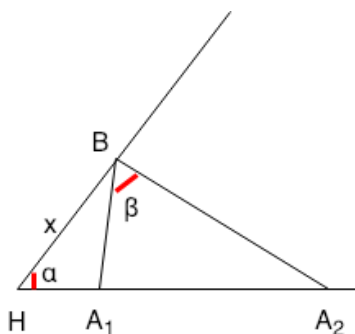
**Задача 1.1.5. (20 баллов)**

Дан некоторый острый угол  $\alpha = 60^\circ$ . На одной из его сторон отмечены точки  $A_1$  и  $A_2$ , на другой стороне отмечена точка  $B$ .

Вершина угла –  $H$ . Известно, что  $HA_1 = 2$ ,  $A_1A_2 = 8$ . При какой величине отрезка  $HB$  величина острого угла между прямыми  $A_1B$  и  $A_2B$  будет максимальна? Ответ введите с точностью до десятитысячных.

**Решение**

Обозначим  $\alpha = \angle BAA_2 = 60^\circ$ ,  $\beta = \angle A_1BA_2$ ,  $x = BH$ .



По теореме косинусов выразим  $A_1B$  и  $A_2B$  и подставим значения  $A_1H = 2$  и  $A_1A_2 = 8$  из условия:

$$A_1B^2 = x^2 + A_1H^2 - 2xA_1H \cdot \cos \alpha = x^2 + 4 - 2x,$$

$$A_2B^2 = x^2 + A_2H^2 - 2xA_2H \cdot \cos \alpha = x^2 + 100 - 10x.$$

Рассмотрим  $\triangle A_1BA_2$ , по теореме косинусов:

$$A_1A_2^2 = A_1B^2 + A_2B^2 - 2A_1B \cdot A_2B \cos \beta.$$

Выразим  $\cos \beta$ :

$$\begin{aligned} \cos \beta &= \frac{A_1B^2 + A_2B^2 - A_1A_2^2}{2A_1B \cdot A_2B} = \frac{x^2 + 4 - 2x + x^2 + 100 - 10x - 64}{2\sqrt{x^2 + 4 - 2x} \cdot \sqrt{x^2 + 100 - 10x}} = \\ &= \frac{x^2 - 6x + 20}{\sqrt{x^2 + 4 - 2x} \cdot \sqrt{x^2 + 100 - 10x}}. \end{aligned}$$

Для максимизации острого угла  $A_1BA_2$  требуется, найти  $x$ , при котором достигается  $\min(\cos \beta)$ . Решим уравнение  $(\cos \beta)'_x = 0$  и найдем точку минимума.  $x = 2\sqrt{5} \approx 4.472135955$ .

**Ответ:** 4.472135955.

## 1.2. Первая попытка. Задачи 10-11 класса.

### Задача 1.2.1. (10 баллов)

Гоша взял у друга 11 гаек М6 (ГОСТ 5916-70) и положил в карман рюкзака. Согласно ГОСТу 1 гайка М6 весит 1.254 грамма. И вот незадача, придя домой, Гоша насчитал в кармане 12 внешне одинаковых гаек! Одна из них была из того набора, что когда-то был куплен на блошином рынке, и, по его личному опыту, такие гайки имеют меньший вес, около грамма, а также сами по себе более низкого качества, менее прочные.

У Гошиного папы есть весы, состоящие из двух больших чаш на двух концах рычага. За какое минимальное количество взвешиваний можно найти ту самую низкокачественную гайку?

#### Решение

Если мы имеем  $n$  внешне одинаковых объектов, после одного взвешивания останется в худшем случае  $\lceil n/2 \rceil$  объектов, среди которых есть отличающийся по весу. Таким образом, если  $n = 12$ , то, в худшем случае, понадобятся 3 взвешивания  $12 \Rightarrow 6 \Rightarrow 3 \Rightarrow 1$ .

Ответ: 3.

### Задача 1.2.2. (10 баллов)

Витя приклеивал цифры номера квартиры на дверь. Этот номер состоит из трех цифр. В процессе приклеивания Вите пришла необычная мысль, что если в номере квартиры поменять местами две последние цифры и сложить получившееся число с исходным, то получится номер его школы! Юноша учился в школе 1187. Найдите все такие номера квартир, и если Витя живет в квартире с наименьшим из них, то в какой квартире он живет?

#### Решение

Пусть номер квартиры равен  $\overline{abc} = 100 \cdot a + 10 \cdot b + c$ , где  $a$ ,  $b$  и  $c$  – цифры числа. Число с перевернутыми двумя последними цифрами при этом будет равно  $\overline{acb} = 100 \cdot a + 10 \cdot c + b$ . Затем решим уравнение в целых числах

$$100 \cdot a + 10 \cdot b + c + 100 \cdot a + 10 \cdot c + b = 1187,$$

$$200 \cdot a + 11 \cdot b + 11 \cdot c = 1187$$

с учетом ограничений на то, что  $a$ ,  $b$  и  $c$  – цифры в десятичной системе счисления.

$$200 \cdot a + 11 \cdot (b + c) = 200 \cdot 5 + 11 \cdot 17.$$

Таким образом, возможные номера квартир 589 и 598. Выбираем наименьший – 589.

Ответ: 589.

**Задача 1.2.3. (10 баллов)**

В химико-биологическом классе 25 учащихся. Для дежурства по школе всегда наугад выбирают двоих. Вероятность того, что оба дежурных окажутся мальчиками, равна  $3/25$ . Сколько в классе девочек?

**Решение**

Пусть  $N$  – количество учащихся в классе,  $A$  – количество юношей в классе. Вероятность того, что оба дежурных окажутся мальчиками составляет

$$P = \frac{A}{N} \cdot \frac{A-1}{N-1}.$$

Подставим из условия  $N = 25$ ,  $P = 3/25$  и получим следующее уравнение:

$$\frac{A}{25} \cdot \frac{A-1}{24} = \frac{3}{25},$$

$$A^2 - A - 3 \cdot 24 = 0.$$

Уравнение имеет только один положительный корень  $A = 9$ . Значит, девочек в классе  $25 - 9 = 16$ .

**Ответ:** 16.

**Задача 1.2.4. (10 баллов)**

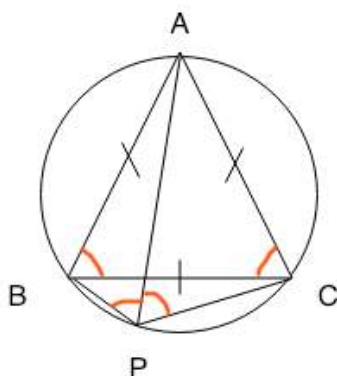
На уроке геометрии нарисовали окружность. На дуге  $BC$  этой окружности, описанной около равностороннего треугольника  $ABC$ , взята произвольная точка  $P$ . Выразите отрезок  $AP$  через отрезки  $BP$  и  $CP$ .

Укажите длину  $AP$ , если  $BP = 3$ ,  $CP = 4$ . Ответ введите с точностью до десяти тысячных.

**Решение**

Так как  $\triangle ABC$  – равносторонний, то  $\angle BAC = \angle ABC = \angle ACB = 60^\circ$  и  $AB = AC$ .  $\angle ABC = \angle APC$ , так как опираются на  $\sphericalangle AC$ .  $\angle ACB = \angle APB$ , так как опираются на  $\sphericalangle AB$ . Таким образом,  $\angle APB = \angle APC = 60^\circ$ . По теореме косинусов

$$\begin{cases} AB^2 = AP^2 + BP^2 - 2 \cdot AP \cdot BP \cdot \cos(\angle APB), \\ AC^2 = AP^2 + CP^2 - 2 \cdot AP \cdot CP \cdot \cos(\angle APC). \end{cases}$$



Так как  $AB = AC$  и  $\angle APC = \angle APB = 60^\circ$ , получим:

$$\begin{aligned} AP^2 + BP^2 - AP \cdot BP &= AP^2 + CP^2 - AP \cdot CP, \\ BP^2 - CP^2 &= AP \cdot (BP - CP), \\ AP &= BP - CP. \end{aligned}$$

Подставим значения  $BP = 3$  и  $CP = 4$  и получим  $AP = 7$ .

**Ответ:** 7.

### Задача 1.2.5. (10 баллов)

Маша и Андрей, будущие математики, развлекались на перемене. Маша написала на доске 4 различных натуральных числа. Андрей выписал значения наибольших общих делителей для каждой из шести пар чисел. Получилось, что для одной из пар НОД равен 1, для другой — 2, для третьей — 3, для четвертой — 4, для пятой — 5, а для шестой —  $X$ . Найдите наименьшее возможное значение  $X$ ?

#### Решение

Пусть на доске записаны 4 числа  $a, b, c$  и  $d$ . Пусть  $\text{НОД}(a, b) = 2$ , тогда  $\text{НОД}(c, d) = 4$  быть не может, так как НОД всех пар будет кратным 2. Значит, пусть  $\text{НОД}(a, c) = 4$  и тогда  $d$  будет нечётным числом.

Исходя из записанных равенств, можно выписать следующие:

$$a = 4 \cdot a_4 = 2 \cdot a_2$$

$$b = 2 \cdot b_2$$

$$c = 4 \cdot c_4$$

При этом  $\text{НОД}(a_2, b_2) = 1$  и  $\text{НОД}(a_4, c_4) = 1$ .

Очевидно также, что  $\text{НОД}(b, c) = 2 \cdot x$  — то есть это последняя искомая пара. Попробуем подобрать значения  $a, b, c$  и  $d$  так, чтобы они удовлетворяли всем равенствам и из НОД оставшихся пар равнялись указанным значениям. Например,  $a = 4$ ,  $b = 10$ ,  $c = 12$  и  $d = 15$ . Таким образом, наименьшее возможное значение равно 2.

**Ответ:** 2.

**Задача 1.2.6. (10 баллов)**

Женя загадал некоторое число: его наименьший делитель (не равный 1) на 77 меньше наибольшего делителя (не равного самому числу). Чему равно это число? Укажите наименьшее из возможных.

**Решение**

Пусть загаданное число  $X = A \cdot B$ , где  $A$  – наименьший делитель,  $B$  – наибольший. Тогда  $A + 77 = B$ .  $X = A \cdot (A + 77) \rightarrow \min$ . Так как  $A \geq 1$ , то минимум достигается при  $A = 2$ . Таким образом,  $X = 2 \cdot 79 = 158$ .

**Ответ:** 158.

**Задача 1.2.7. (10 баллов)**

У Лады на прикроватной тумбочке стоят часы с циферблатом. Они показывают текущее время суток от 00.00.00 до 23.59.59. Однако, сосед Дима решил перепрошить часы, и теперь, если на часах должны загореться ровно четыре цифры 3, циферблат перестает гореть. Сколько времени в течение суток часы не показывают время, если всё остальное время они работают корректно? Ответ укажите в секундах.

**Решение**

Каждая комбинация цифр на циферблате отображается ровно 1 секунду, следовательно в задаче требуется найти количество соответствующих комбинаций. В группе цифр, отображающей часы может встретиться только 0 или 1 тройка. В остальных – 0, 1 или 2. Четыре тройки можно получить из следующих комбинаций:  $X_1 = Q([0] : [2] : [2])$ ,  $X_2 = Q([1] : [1] : [2])$ ,  $X_3 = Q([1] : [2] : [1])$ .  $Q$  – количество комбинаций в соответствии с количеством троек в группах цифр циферблата, соответствующих часам, минутам, секундам.  $X_1 = 21 \cdot 1 \cdot 1$ .  $X_2 = 3 \cdot 14 \cdot 1$ .  $X_3 = 3 \cdot 1 \cdot 14$ .  $X = X_1 + X_2 + X_3 = 21 + 42 + 42 = 105$ .

**Ответ:** 105.

**Задача 1.2.8. (10 баллов)**

Дано уравнение  $x^3 + y^3 = 4(x^2y + xy^2 + 1)$ . Сколько различных решений в целых числах оно имеет?

Если решений бесконечное множество, введите -1.

**Решение**

$$x^3 + y^3 = 4(x^2y + xy^2 + 1),$$

$$(x + y)(x^2 - xy + y^2) - 4xy(x + y) = 4,$$



$$(x + y)(x^2 - 5xy + y^2) = 4.$$

В левой части множители могут принимать только следующие пары значений:  $(-4, 1)$ ,  $(-2, -2)$ ,  $(-1, -4)$ ,  $(1, 4)$ ,  $(2, 2)$ ,  $(4, 1)$ . Все 6 систем уравнений не дают целых корней.

Ответ: 0.

### Задача 1.2.9. (10 баллов)

Дан некоторый острый угол  $\alpha = 60^\circ$ . На одной из его сторон отмечены точки  $A_1$  и  $A_2$ , на другой стороне отмечена точка  $B$ .

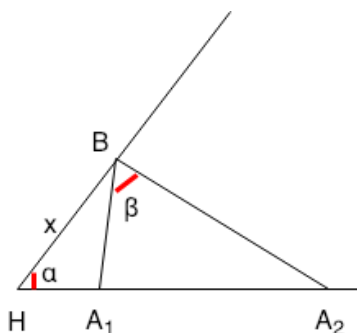
Вершина угла —  $H$ . Известно, что  $HA_1 = 2$ ,  $A_1A_2 = 8$ .

При какой величине отрезка  $HB$  величина острого угла между прямыми  $A_1B$  и  $A_2B$  будет максимальна?

Ответ введите с точностью до десятитысячных.

### Решение

Обозначим  $\alpha = \angle BAA_2 = 60^\circ$ ,  $\beta = \angle A_1BA_2$ ,  $x = BH$ .



По теореме косинусов выразим  $A_1B$  и  $A_2B$  и подставим значения  $A_1H = 2$  и  $A_1A_2 = 8$  из условия:

$$A_1B^2 = x^2 + A_1H^2 - 2xA_1H \cdot \cos \alpha = x^2 + 4 - 2x,$$

$$A_2B^2 = x^2 + A_2H^2 - 2xA_2H \cdot \cos \alpha = x^2 + 100 - 10x.$$

Рассмотрим  $\triangle A_1BA_2$ , по теореме косинусов:

$$A_1A_2^2 = A_1B^2 + A_2B^2 - 2A_1B \cdot A_2B \cos \beta.$$

Выразим  $\cos \beta$ :

$$\begin{aligned} \cos \beta &= \frac{A_1B^2 + A_2B^2 - A_1A_2^2}{2A_1B \cdot A_2B} = \frac{x^2 + 4 - 2x + x^2 + 100 - 10x - 64}{2\sqrt{x^2 + 4 - 2x} \cdot \sqrt{x^2 + 100 - 10x}} = \\ &= \frac{x^2 - 6x + 20}{\sqrt{x^2 + 4 - 2x} \cdot \sqrt{x^2 + 100 - 10x}}. \end{aligned}$$

Для максимизации острого угла  $A_1BA_2$  требуется, найти  $x$ , при котором достигается  $\min(\cos \beta)$ . Решим уравнение  $(\cos \beta)'_x = 0$  и найдем точку минимума.  $x = 2\sqrt{5} \approx 4.472135955$ .

**Ответ:** 4.472135955.

### **Задача 1.2.10. (10 баллов)**

Коптер летит над поверхностью огромного поля.

В какой-то момент времени он оказывается в точке  $(0, 3, 6)$  в заданной ортогональной системе координат с осями  $Ox, Oy$  и  $Oz$ .

Найдите расстояние от коптера до земли, если в той же системе координат поле можно считать плоскостью, заданной уравнением  $2x + 4y - 4z - 6 = 0$ . Ответ укажите с точностью до десятитысячных.

#### **Решение**

Расстояние между точкой с координатами  $(x_0, y_0, z_0)$  и плоскостью, задаваемой уравнением  $Ax + By + Cz + D = 0$  вычисляется по формуле:

$$S = \frac{|Ax_0 + By_0 + Cz_0 + D|}{\sqrt{A^2 + B^2 + C^2}}$$

Для заданных точки и плоскости

$$S = \frac{|2 \cdot 0 + 4 \cdot 3 - 4 \cdot 6 - 6|}{\sqrt{2^2 + 4^2 + 4^2}} = 3.$$

**Ответ:** 3.

## **1.3. Вторая попытка. Задачи 9 класса.**

### **Задача 1.3.1. (20 баллов)**

Полина прячет в кулаке от 1 до 4 гвоздей. Валера пытается угадать, сколько их. Для этого он задаёт вопросы, на которые Полина может отвечать "да" и "нет". За какое минимальное число вопросов Валера может угадать количество спрятанных гвоздей.

#### **Решение**

За один вопрос можно сократить перебираемое множество вдвое. Таким образом, достаточно двух вопросов.

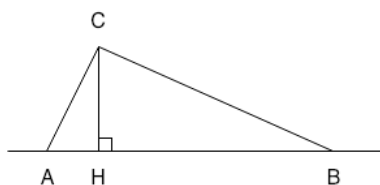
**Ответ:** 2.

**Задача 1.3.2. (20 баллов)**

В стране  $X$  есть три города –  $A, B, C$ . Известно, что расстояние между  $AB = 25$ ,  $CB = 24$ ,  $AC = 7$ . Города  $A$  и  $B$  лежат на прямолинейной границе страны, а все остальные участки границы страны пренебрежительно далеки. Найдите кратчайший путь из города  $C$  до границы страны  $X$ . Ответ укажите с точностью до десятитысячных.

**Решение**

Наикратчайший путь равен длине перпендикуляра, опущенного из вершины  $C$  к прямой  $AB$ .



По длинам сторон очевидно, что  $\triangle ABC$  – прямоугольный, с прямым углом в вершине  $C$  ( $7^2 + 24^2 = 25^2$ ). Так, согласно свойству прямоугольного треугольника,  $CH = AC \cdot BC / AB = 6.72$ .

**Ответ:** 6.72.

**Задача 1.3.3. (20 баллов)**

Коля – очень любознательный юноша. Он решил провести исследование. Для различных действительных чисел  $a$  он решил найти такое наибольшее целое число  $x$ , чтобы выполнялось следующее:

1.  $a$  лежит в интервале  $(1, 2)$ ,
2.  $a^2$  лежит в  $(2, 3)$ ,
3.  $a^3$  лежит в  $(3, 4)$ ,
4. и так далее до показателя степени  $x$ .

Помогите Коле выяснить, каким же может быть максимальное значение числа  $x$ , при котором существует хотя бы одно значение  $a$ , удовлетворяющее условиям?

**Решение**

В данной задаче необходимо найти такое натуральное максимальное  $n$ , при котором  $\exists a \forall x \in \mathbb{N} : x < n \Rightarrow x < a^x < x + 1$ . Выпишем границы интервалов  $(\sqrt[x]{x}, \sqrt[x]{x+1})$  для нескольких первых значений  $x$ .

$x$	Левая граница $a$	Правая граница $a$
1	1	2
2	$\sqrt[2]{2} = 1.41421356$	$\sqrt[2]{3} = 1.73205081$
3	$\sqrt[3]{3} = 1.44224957$	$\sqrt[3]{4} = 1.58740105$
4	$\sqrt[4]{4} = 1.41421356$	$\sqrt[4]{5} = 1.49534878$
5	$\sqrt[5]{5} = 1.37972966$	$\sqrt[5]{6} = 1.43096908$
...	...	...

По таблице видно, что  $\sqrt[5]{6} < \sqrt[3]{3}$ . Таким образом,  $n = 4$  – наибольшее значение, удовлетворяющее условию задачи.

**Ответ:** 4.

### **Задача 1.3.4. (20 баллов)**

Найдите сумму коэффициентов после раскрытия скобок у выражения

$$(x^2 - 3x + 1)^{100}.$$

#### **Решение**

Постепенно раскрывая скобки найдем сумму коэффициентов полинома  $(x^2 - 3x + 1)^n$  для  $n$ , начиная с 1.

$n$	Сумма коэффициентов
1	-1
2	1
3	-1
4	1
...	...

Подметим, что при нечётном  $n$  сумма коэффициентов равна  $-1$ , а при чётном  $-1$ .

**Ответ:** 1.

### **Задача 1.3.5. (20 баллов)**

На доске записаны 5 чисел: сначала некоторое рациональное  $a = n/y$  ( $n$  и  $y$  натуральные взаимно простые числа), затем  $x$  и далее  $x + 2$ ,  $x + 3$  и  $x + 4$ . При каком наименьшем значении  $a$  произведение всех пяти чисел всегда будет натуральным для любого натурального  $x$ ? В ответе напишите целое число  $y$ .

#### **Решение**

В данной задаче надо проверить делимость произведения чисел.  $x \cdot (x + 2) \cdot (x + 3) \cdot (x + 4)$  гарантированно делится на 2 и на 3, так как содержит произведение последовательно идущих 3 чисел. Делимость на остальные числа при любом  $x$  гарантировать нельзя.

Ответ: 6.

## 1.4. Вторая попытка. Задачи 10-11 класса.

### Задача 1.4.1. (10 баллов)

Полина прячет в кулаке от 1 до 4 гвоздей. Валера пытается угадать, сколько их. Для этого он задаёт вопросы, на которые Полина может отвечать "да" и "нет". За какое минимальное число вопросов Валера может угадать количество спрятанных гвоздей.

#### Решение

За один вопрос можно сократить перебираемое множество вдвое. Таким образом, достаточно двух вопросов.

Ответ: 2.

### Задача 1.4.2. (10 баллов)

Мотоциклист поднимается на холм. Его движение в ортогональной системе координат  $xOy$  можно описать законом  $y = ax^2 + bx + c$ , где  $a$  и  $b$  – некоторые неизвестные постоянные коэффициенты. Известно, что во время своего движения мотоциклист побывал в точках с координатами  $(0, 1)$ ,  $(1, 2)$ ,  $(2, 1)$ . Найдите координаты вершины холма. Ответ укажите в формате " $(x, y)$ " где  $x$  и  $y$  – значения абсциссы и ординаты с точностью до десятичных.

#### Решение

Подставим точки в уравнение и решим систему:

$$\begin{cases} 1 = a \cdot 0^2 + b \cdot 0 + c, \\ 2 = a \cdot 1^2 + b \cdot 1 + c, \\ 1 = a \cdot 2^2 + b \cdot 2 + c; \end{cases}$$

$$\begin{cases} c = 1, \\ a + b = 1, \\ 2a + b = 0; \end{cases}$$

$$\begin{cases} a = -1, \\ b = 2, \\ c = 1. \end{cases}$$

Уравнение параболы с вычисленными коэффициентами  $y = -x^2 + 2x + 1$ . Найдём абсциссу точки перегиба:

$$y'_x = 0 \Leftrightarrow -2x + 2 = 0 \Leftrightarrow x = 1.$$

При  $x = 1$   $y = 2$ .

**Ответ:** (1, 2).

### Задача 1.4.3. (10 баллов)

Для тестирования новой программы компьютер выбирает случайное действительное число  $A$  из отрезка  $[1, 2]$  и заставляет программу решать уравнение  $3x + A = 0$ . Учтите, что распределение случайной величины равномерное. Найдите вероятность того, что корень этого уравнения меньше, чем  $-0.4$ . Ответ укажите с точностью до десятитысячных.

#### Решение

Выразим корень уравнения:  $x = -A/3$ . Таким образом,  $x \in [-2/3, -1/3]$ . Значит

$$P(x \in [-2/3, -0.4]) = \frac{-0.4 + 2/3}{1/3} = 2 - 1.2 = 0.8.$$

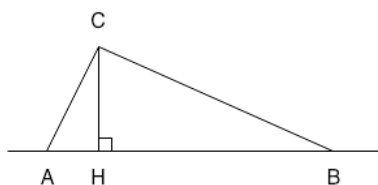
**Ответ:** 0.8.

### Задача 1.4.4. (10 баллов)

В стране  $X$  есть три города –  $A, B, C$ . Известно, что расстояние между  $AB = 25$ ,  $CB = 24$ ,  $AC = 7$ . Города  $A$  и  $B$  лежат на прямолинейной границе страны, а все остальные участки границы страны пренебрежительно далеки. Найдите кратчайший путь из города  $C$  до границы страны  $X$ . Ответ укажите с точностью до десятитысячных.

#### Решение

Наикратчайший путь равен длине перпендикуляра, опущенного из вершины  $C$  к прямой  $AB$ .



По длинам сторон очевидно, что  $\triangle ABC$  – прямоугольный, с прямым углом в вершине  $C$  ( $7^2 + 24^2 = 25^2$ ). Так, согласно свойству прямоугольного треугольника,  $CH = AC \cdot BC / AB = 6.72$ .

**Ответ:** 6.72.

**Задача 1.4.5. (10 баллов)**

Коля – очень любознательный юноша. Он решил провести исследование. Для различных действительных чисел  $a$  он решил найти такое наибольшее целое число  $x$ , чтобы выполнялось следующее:

1.  $a$  лежит в интервале  $(1, 2)$ ,
2.  $a^2$  лежит в  $(2, 3)$ ,
3.  $a^3$  лежит в  $(3, 4)$ ,
4. и так далее до показателя степени  $x$ .

Помогите Коле выяснить, каким же может быть максимальное значение числа  $x$ , при котором существует хотя бы одно значение  $a$ , удовлетворяющее условиям?

**Решение**

В данной задаче необходимо найти такое натуральное максимальное  $n$ , при котором  $\exists a \forall x \in \mathbb{N} : x < n \Rightarrow x < a^x < x + 1$ . Выпишем границы интервалов  $(\sqrt[x]{x}, \sqrt[x]{x+1})$  для нескольких первых значений  $x$ .

$x$	Левая граница $a$	Правая граница $a$
1	1	2
2	$\sqrt[2]{2} = 1.41421356$	$\sqrt[2]{3} = 1.73205081$
3	$\sqrt[3]{3} = 1.44224957$	$\sqrt[3]{4} = 1.58740105$
4	$\sqrt[4]{4} = 1.41421356$	$\sqrt[4]{5} = 1.49534878$
5	$\sqrt[5]{5} = 1.37972966$	$\sqrt[5]{6} = 1.43096908$
...	...	...

По таблице видно, что  $\sqrt[5]{6} < \sqrt[3]{3}$ . Таким образом,  $n = 4$  – наибольшее значение, удовлетворяющее условию задачи.

**Ответ:** 4.

**Задача 1.4.6. (10 баллов)**

На доске записаны 5 чисел: сначала некоторое рациональное  $a = n/y$  ( $n$  и  $y$  натуральные взаимно простые числа), затем  $x$  и далее  $x + 2$ ,  $x + 3$  и  $x + 4$ . При каком наименьшем значении  $a$  произведение всех пяти чисел всегда будет натуральным для любого натурального  $x$ ? В ответе напишите целое число  $y$ .

**Решение**

В данной задаче надо проверить делимость произведения чисел  $x \cdot (x+2) \cdot (x+3) \cdot (x+4)$  гарантированно делится на 2 и на 3, так как содержит произведение последовательно идущих 3 чисел. Делимость на остальные числа при любом  $x$  гарантировать нельзя.

**Ответ:** 6.

**Задача 1.4.7. (10 баллов)**

Двое бросают монету: один бросил её 8 раз, другой – 9 раз. Чему равна вероятность того, что у второго монета упала орлом большее число раз, чем у первого?

Монета никогда не падает на ребро.

Ответ укажите с точностью до десятитысячных.

**Решение**

Первый может выбросить 9 различных наборов (последовательность не учитывается). Если в наборе  $x$  орлов, то количество возможных вариантов второго игрока, удовлетворяющих условию  $(9 - x)$ . Всего возможных способов выкинуть наборы у обоих игроков:  $9 \cdot 10 = 90$ .

$$P = \frac{\sum_{x=0}^8 9 - x}{90} = 0.5.$$

**Ответ:** 0.5.

**Задача 1.4.8. (10 баллов)**

Найдите сумму коэффициентов после раскрытия скобок у выражения

$$(x^2 - 3x + 1)^{100}.$$

**Решение**

Постепенно раскрывая скобки найдем сумму коэффициентов полинома  $(x^2 - 3x + 1)^n$  для  $n$ , начиная с 1.

$n$	Сумма коэффициентов
1	-1
2	1
3	-1
4	1
...	...

Подметим, что при нечётном  $n$  сумма коэффициентов равна  $-1$ , а при чётном  $-1$ .

**Ответ:** 1.

**Задача 1.4.9. (10 баллов)**

Самолет взлетает с авианосца. Вектор нормали к поверхности взлетной полосы имеет координаты  $(4, 0, 3)$ . Направляющий вектор траектории полета



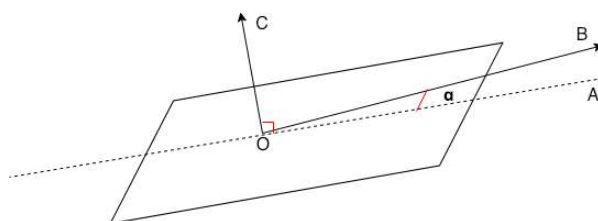
самолета –  $(5, 12, 0)$  в той же ортогональной системе координат. Найдите СИНОС угла, под которым взлетел самолет относительно взлетной полосы.

Введите ответ с точностью до десятитысячных.

*Решение*

$$\cos \alpha = \frac{|(\vec{OB}, \vec{OC})|}{|\vec{OB}| \cdot |\vec{OC}|} = \frac{20}{\sqrt{25 + 144}\sqrt{16 + 9}} = \frac{4}{13},$$

$$\sin \alpha = \sqrt{1 - \cos^2 \alpha} = 0.9514859.$$



Ответ: 0.9514859136.

### **Задача 1.4.10. (10 баллов)**

Соня, Андрей и Егор живут в домах  $A, B, C$  соответственно. Эти дома соединены прямыми улицами – Садовой, Огородной и Персиковой. Известно, что Садовая и Персиковая улицы пересекаются у дома Сони под углом 45 градусов, Персиковая и Огородная – под углом 60 градусов у дома Егора и, наконец, Огородная и Садовая пересекаются под окнами у Андрея. Равноудаленно от домов  $B$  и  $C$  внутри треугольника  $ABC$  построили магазин. Известно, что прямая улица, которая соединяет магазин и дом Егора, пересекается с Персиковой под углом 15 градусов. Между домом Сони и магазином также есть прямая улица. Под каким углом она пересекается с Садовой?

Ответ укажите в градусах с точностью до десятитысячных.

*Решение*

Обозначим искомый угол буквой  $\alpha$ .



$\angle BAC = 45^\circ \Rightarrow \angle OAC = 45^\circ - \alpha$ .  $\angle OCA = 15^\circ \Rightarrow \angle AOC = 120^\circ + \alpha$ .  
 $\angle OCB = \angle OBC = 60^\circ - 15^\circ = 45^\circ \Rightarrow \angle BOC = 90^\circ$ .  $\angle ABO = 30^\circ$ . По теореме синусов:

$$\frac{\sin \alpha}{OB} = \frac{\sin 30^\circ}{AO} \Rightarrow AO = \frac{OB}{2 \sin \alpha}$$

$$\frac{\sin 15^\circ}{OA} = \frac{\sin(45^\circ - \alpha)}{OC} \Rightarrow AO = \frac{OC \sin 15^\circ}{\sin(45^\circ - \alpha)}$$

Так как  $OB = OC$ ,

$$\sin(45^\circ - \alpha) = 2 \sin \alpha \sin 15^\circ \Rightarrow \alpha = 30^\circ.$$

Ответ: 30.

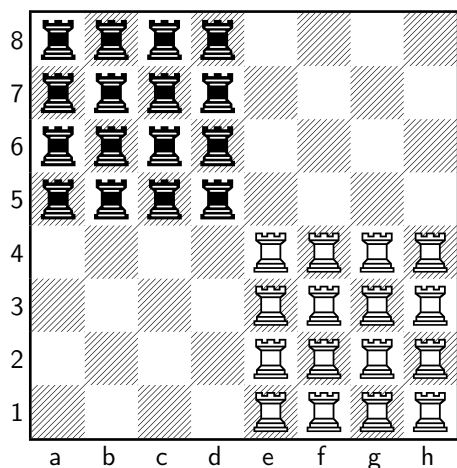
## 1.5. Третья попытка. Задачи 9 класса.

### Задача 1.5.1. (20 баллов)

Пусть на шахматной доске размера  $8 \times 8$  расставили  $n$  белых и столько же черных ладей так, что они не бьют друг друга. Найдите такое максимальное  $n$ , при котором это возможно.

#### Решение

Необходимо расставить ладьи одного цвета так, что на одной горизонтали с каждой из них было еще 3 и на одной вертикали 3 другие. Таким образом, в каждой строке или столбце ровно 4 ладьи одного цвета. Например:



Ответ: 16.

### Задача 1.5.2. (20 баллов)

На планете  $Y$  все жители говорят на языке, состоящем из трех букв. При этом все слова в этом языке не длиннее 6 символов и не короче 3. Сколько слов можно составить в языке планеты  $Y$ ?

*Решение*

Количество слов из 3 букв длиной  $n$   $W_n = 3^n$ . Так  $W = 3^3 + 3^4 + 3^5 + 3^6 = 1080$ .

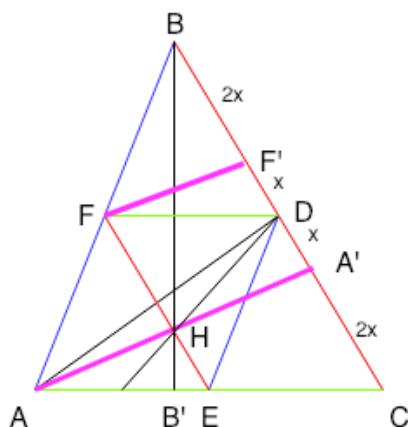
Ответ: 1080.

### Задача 1.5.3. (20 баллов)

Дан треугольник  $ABC$  и  $H$  — точка пересечения высот этого треугольника. Пусть  $D$  — середина отрезка  $BC$ ,  $E$  — середина отрезка  $AC$ . Кроме того, медианы треугольника  $AED$  пересекаются в точке  $H$ . Найдите градусную меру угла  $\angle ABC$ .

Ответ укажите с точностью до десятитысячных.

*Решение*



Рассмотрим  $\triangle ABC$ :  $DE, EF, DF$  – медианы,  $AA', BB'$  – высоты.  $BD = DC$ . В  $\triangle ADE$ : медианы пересекаются в точке  $H$ , которая разделяет их в отношении  $2 : 1$ . Из подобия треугольников очевидно, что  $DA' : A'C = 1 : 2$ .

Пусть  $AD = x$ ,  $A'C = 2x$ .  $FH = 2 \cdot (HE + HE/2) - HE = 2HE = 2x$ . По свойству параллелограмма, образованного медианами  $FE$  и  $BC$ , перпендикулярами  $AA'$  и  $FF'$ :  $FH = F'A' = 2x$  и  $FF' = HA'$ ,  $F'D = x$ . Также из подобия треугольников  $BF' = 2x$ ,  $FF' = HA' = AH$ .

Рассмотрим  $\triangle BA'H \sim \triangle AHE$  (по 2 углам).  $A'B : A'H = A'H : HE \Rightarrow A'H = \sqrt{A'B \cdot HE} = \sqrt{4x^2} = 2x = EF' = BF'$ .

Так,  $\triangle BF'F$  – прямоугольный равнобедренный, и угол при основании  $\angle FBF' = 45^\circ$ .

**Ответ:** 45.

### **Задача 1.5.4. (20 баллов)**

Артур и Саша играли в игру – по очереди выписывали натуральные числа на бумагу. В итоге оказалось, что на бумаге выписано 15 чисел, причем, наименьшее из чисел можно представить как  $x + 1$ ,  $x > 1$ , а все остальные числа – последовательность  $(1 + x^n)$ , где  $n$  – натуральный показатель степени, изменяющийся от 2 до 15. Артуру показалось, что выписанных чисел слишком много и он зачеркнул часть из них таким образом, чтобы все оставшиеся на бумаге числа были взаимно простыми. Какое наименьшее количество чисел мог зачеркнуть Артур?

#### **Решение**

Все натуральные числа вида  $(1 + x^n)$ , где  $n$  – нечетное число, делятся нацело на  $(1 + x)$ .

Таким образом, вычеркнем все числа, где  $n = 1, 3, 5, 7, 9, 11$  и  $13$ . 15 оставляем.

Далее заменим  $x^2$  на  $y$ . По аналогии вычеркнем числа, где  $n = 2, 6, 10, 14$  оставляем.

Далее заменим  $x^3$  на  $z$ .  $n = 3, 9$  – вычеркнуты. 15 оставляем.

Далее заменим  $x^4$  на  $w$ . По аналогии вычеркнем число, где  $n = 4$ . 12 оставляем.

Далее заменим  $x^5$  на  $v$ .  $n = 5$  вычеркнуто. 15 оставляем.

Далее нет смысла перебирать, так как  $3n > 15$ . Оставшиеся числа взаимнопростые. Таким образом, минимально мы вычеркнули 11 чисел (1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 13).

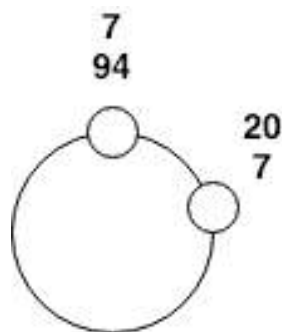
**Ответ:** 11.

### **Задача 1.5.5. (20 баллов)**

Ира и Паша расставляют стулья вокруг круглого стола. После того, как все стулья были расставлены, ребята решили их пересчитать – они начали ходить по кругу

в одном направлении, но начиная с разных стульев. Известно, что стул, который Паша посчитал седьмым, у Иры оказался под двадцатым номером, а тот стул, который Ира посчитала седьмым, у Паши был 94 м. Сколько стульев было расставлено вокруг стола?

*Решение*



Считаем по обоим дугам количество стульев, включая один из двух крайних:  
 $20 - 7 + 94 - 7 = 100$ .

Ответ: 100.

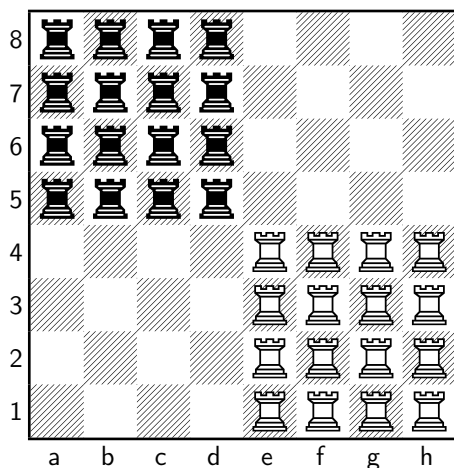
## 1.6. Третья попытка. Задачи 10-11 класса.

### Задача 1.6.1. (10 баллов)

Пусть на шахматной доске размера  $8 \times 8$  расставили  $n$  белых и столько же черных ладей так, что они не бьют друг друга. Найдите такое максимальное  $n$ , при котором это возможно.

*Решение*

Необходимо расставить ладьи одного цвета так, что на одной горизонтали с каждой из них было еще 3 и на одной вертикали 3 другие. Таким образом, в каждой строке или столбце ровно 4 ладьи одного цвета. Например:



Ответ: 16.

**Задача 1.6.2. (10 баллов)**

Найдите такое значение  $a > 1$ , при котором уравнение  $a^x = \log_a x$  имеет ровно один корень. Ответ укажите с точностью до десятичных.

**Решение**

Обе функции в левой и правой частях уравнения выпуклые, единственный корень у уравнения возникает тогда, когда  $y = a^x$  касается прямой  $y = x$ , то есть  $f(x_0) = x_0$  и  $f'(x_0) = a^{x_0} \ln a = 1$ . Отсюда  $x_0 = \frac{1}{\ln a}$ , т.е.  $e = a^{\frac{1}{\ln a}} = \frac{1}{\ln a}$ ,  $\ln a = \frac{1}{e}$ ,  $a = e^{\frac{1}{e}} = 1.44466786$ .

Ответ: 1.44466786101.

**Задача 1.6.3. (10 баллов)**

На планете  $Y$  все жители говорят на языке, состоящем из трех букв. При этом все слова в этом языке не длиннее 6 символов и не короче 3. Сколько слов можно составить в языке планеты  $Y$ ?

**Решение**

Количество слов из 3 букв длиной  $n$   $W_n = 3^n$ . Так  $W = 3^3 + 3^4 + 3^5 + 3^6 = 1080$ .

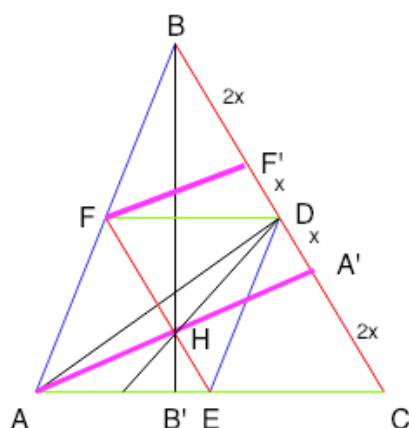
Ответ: 1080.

**Задача 1.6.4. (10 баллов)**

Дан треугольник  $ABC$  и  $H$  — точка пересечения высот этого треугольника. Пусть  $D$  — середина отрезка  $BC$ ,  $E$  — середина отрезка  $AC$ . Кроме того, медианы треугольника  $AED$  пересекаются в точке  $H$ . Найдите градусную меру угла  $\angle ABC$ .

Ответ укажите с точностью до десятичных.

## Решение



Рассмотрим  $\triangle ABC$ :  $DE, EF, DF$  – медианы,  $AA', BB'$  – высоты.  $BD = DC$ . В  $\triangle ADE$ : медианы пересекаются в точке  $H$ , которая разделяет их в отношении  $2 : 1$ . Из подобия треугольников очевидно, что  $DA' : A'C = 1 : 2$ .

Пусть  $AD = x, A'C = 2x$ .  $FH = 2 \cdot (HE + HE/2) = 2HE = 2x$ . По свойству параллелограмма, образованного медианами  $FE$  и  $BC$ , перпендикулярами  $AA'$  и  $FF'$ :  $FH = F'A' = 2x$  и  $FF' = HA', F'D = x$ . Также из подобия треугольников  $BF' = 2x, FF' = HA' = AH$ .

Рассмотрим  $\triangle BA'H \sim \triangle AHE$  (по 2 углам).  $A'B : A'H = A'H : HE \Rightarrow A'H = \sqrt{A'B \cdot HE} = \sqrt{4x^2} = 2x = EF' = BF'$ .

Так,  $\triangle BF'F$  – прямоугольный равнобедренный, и угол при основании  $\angle FBF' = 45^\circ$ .

Ответ: 45.

**Задача 1.6.5. (10 баллов)**

Артур и Саша играли в игру — по очереди выписывали натуральные числа на бумагу. В итоге оказалось, что на бумаге выписано 15 чисел, причем, наименьшее из чисел можно представить как  $x + 1, x > 1$ , а все остальные числа — последовательность  $(1 + x^n)$ , где  $n$  — натуральный показатель степени, изменяющийся от 2 до 15. Артуру показалось, что выписанных чисел слишком много и он зачеркнул часть из них таким образом, чтобы все оставшиеся на бумаге числа были взаимно простыми. Какое наименьшее количество чисел мог зачеркнуть Артур?

## Решение

Все натуральные числа вида  $(1 + x^n)$ , где  $n$  — нечетное число, делятся нацело на  $(1 + x)$ .

Таким образом, вычеркнем все числа, где  $n = 1, 3, 5, 7, 9, 11$  и  $13$ . 15 оставляем.

Далее заменим  $x^2$  на  $y$ . По аналогии вычеркнем числа, где  $n = 2, 6, 10, 14$  оставляем.

Далее заменим  $x^3$  на  $z$ .  $n = 3$ , 9 – вычеркнуты. 15 оставляем.

Далее заменим  $x^4$  на  $w$ . По аналогии вычеркнем число, где  $n = 4$ . 12 оставляем.

Далее заменим  $x^5$  на  $v$ .  $n = 5$  вычеркнуто. 15 оставляем.

Далее нет смысла перебирать, так как  $3n > 15$ . Оставшиеся числа взаимнопростые. Таким образом, минимально мы вычеркнули 11 чисел (1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 13).

**Ответ:** 11.

### **Задача 1.6.6. (10 баллов)**

Паша загадал число  $x$ : это неправильная дробь с натуральным числителем и со знаменателем, равным 9. Далее он вычислил еще три числа умножил  $x$  на 5, на 2 и на 4. Затем округлил эти три числа по правилам округления до целого и сложил между собой, в итоге он получил 120. Каким был числитель у неправильной дроби?

#### **Решение**

Найдем первое значение для перебора:  $\frac{5x+2x+4x}{9} \approx 120 \Leftrightarrow x \approx 98$ . Проверим значение:

$$\left[ \frac{5 \cdot 98}{9} \right] + \left[ \frac{2 \cdot 98}{9} \right] + \left[ \frac{4 \cdot 98}{9} \right] = 54 + 22 + 44 = 120.$$

**Ответ:** 98.

### **Задача 1.6.7. (10 баллов)**

Если из резервуара выливают воду, уровень воды  $H$  в нём меняется в зависимости от времени  $t$  следующим образом:  $H(t) = at^2 + bt + c$ . Пусть  $t_0$  — момент окончания слива. Известно, что в этот момент выполнены равенства  $H(t_0) = H'(t_0) = 0$ . В течение какого времени вода из резервуара будет полностью вылита, если за первый час слилась половина уровня?

Округлите ответ до ближайшего целого.

#### **Решение**

$$H(t_0) = at_0^2 + bt_0 + c = 0,$$

$$H'(t_0) = 2at_0 + b = 0.$$

Выразим  $b$  и  $c$  через  $a$  и  $t_0$ :

$$b = -2at_0,$$

$$c = at_0^2.$$

Пусть  $\tau$  – время достижения половинного уровня.

$$2H(\tau) = H(0),$$



$$\begin{aligned}
 2(a\tau^2 + (-2at_0)\tau + at_0^2) &= at_0^2, \\
 2\tau^2 - 4t_0\tau + t_0^2 &= 0, \\
 D &= 16t_0^2 - 4 \cdot 2t_0^2 = 8t_0^2, \\
 \tau &= \frac{4t_0 \pm 2\sqrt{2}t_0}{2 \cdot 2} = t_0 \pm t_0/\sqrt{2}.
 \end{aligned}$$

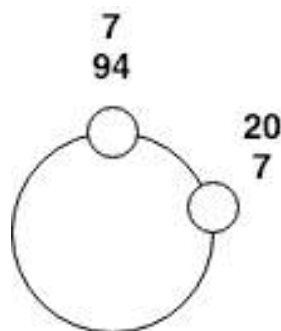
Так как  $\tau < t_0$  из условия, то  $\tau = t_0 \cdot (2 - \sqrt{2})/2$ , то есть  $t_0 = \tau(2 + \sqrt{2})$ , то есть  $[t_0] = [1 \cdot (2 + \sqrt{2})] = 3$ .

**Ответ:** 3.

### Задача 1.6.8. (10 баллов)

Ира и Паша расставляют стулья вокруг круглого стола. После того, как все стулья были расставлены, ребята решили их пересчитать — они начали ходить по кругу в одном направлении, но начиная с разных стульев. Известно, что стул, который Паша посчитал седьмым, у Иры оказался под двадцатым номером, а тот стул, который Ира посчитала седьмым, у Паши был 94 м. Сколько стульев было расставлено вокруг стола?

*Решение*



Считаем по обеим дугам количество стульев, включая один из двух крайних:  $20 - 7 + 94 - 7 = 100$ .

**Ответ:** 100.

### Задача 1.6.9. (10 баллов)

В течение пяти часов 1 сентября Женя наблюдал за воздушными шариками в небе. По мере того, как утренние линейки проходили, шаров в небе становилось все меньше — так, с каждым часом за час пролетало не больше шаров, чем в предыдущий час. Суммарно Женя насчитал 100 шаров, пролетевших в небе мимо его окна. Причем, суммарно за второй и четвертый час Женя увидел не больше шаров, чем за первый и третий. Какое минимальное число шаров Женя мог увидеть суммарно за 1, 3 и 5 часы?

**Решение**

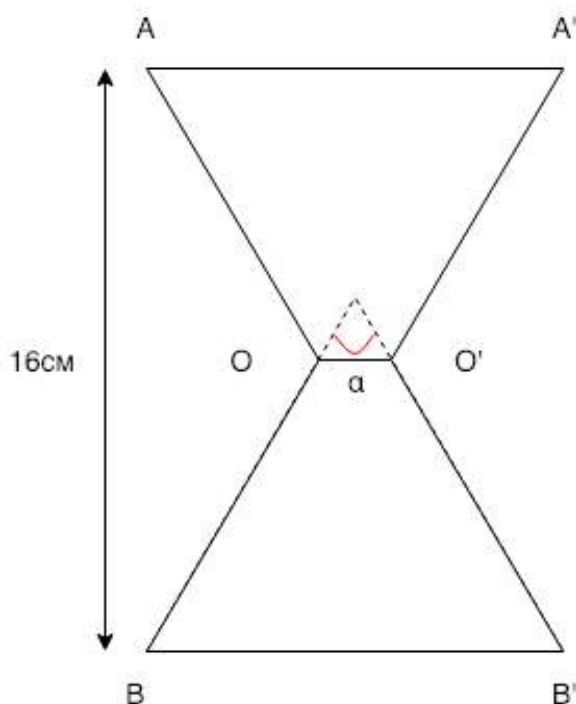
Пусть  $a, b, c, d, e$  — количество шаров в 1, 2, 3, 4 и 5 часы соответственно.  $a \geq b \geq c \geq d \geq e$ ,  $a + b + c + d + e = 100$ ,  $b + d \leq a + c$ , требуется найти  $\min(a + c + e)$ . Данная задача сводится к поиску  $\max(b + d)$ . Так как  $b + d \leq a + c$ , а  $a + b + c + d + e = 100$ , то  $b + d \leq 100/2 = 50$ . Такой случай легко придумать:  $a, b, c, d, e = 25, 25, 25, 25, 0$ . Таким образом,  $\min(a + c + e) = 100 - \max(b + d) = 50$ .

**Ответ:** 50.

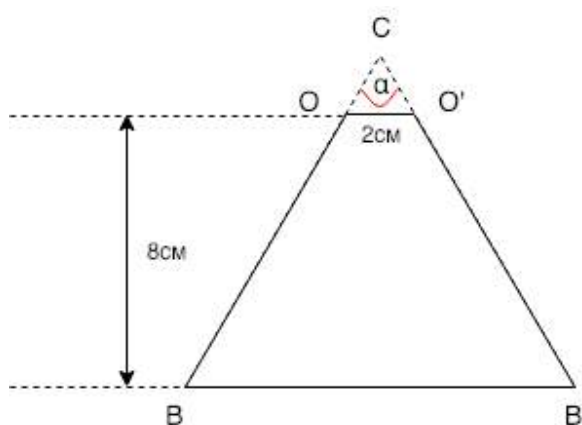
**Задача 1.6.10. (10 баллов)**

На столе стоят песочные часы высоты 16 см, представляющие собой два соединенных усеченных одинаковых конуса. Радиус горлышка (отверстия, через которое сыпется песок) равен 1 см. Тангенс угла раствора конусов равен  $4/3$ . Чему равен объем песочных часов в  $\text{см}^3$ ?

Ответ округлите до ближайшего целого.

**Решение**

Рассмотрим один из конусов.

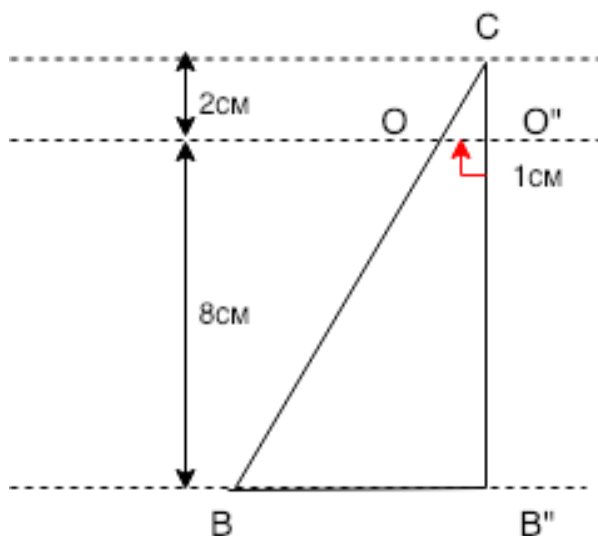


$$\begin{aligned} \operatorname{tg}^2 \alpha + 1 &= \frac{1}{\cos^2 \alpha}, \\ \cos \alpha &= \frac{3}{5}. \end{aligned}$$

Пусть  $OC = O'C = x$ , по теореме косинусов:  $OO'^2 = x^2 + x^2 - 2x \cdot x \cos \alpha = \frac{4x^2}{5}$ ,  
 $x = \sqrt{5}$ .

Опустим перпендикуляр  $CO''$  к  $OO'$ .

$$CO'' = \sqrt{x^2 - OO'^2} = \sqrt{5 - 1} = 2$$



Из подобия треугольников  $BB'' = 5$ . Объем песочных часов, т.е. удвоенный объем усеченного конуса, равен

$$V = \frac{2}{3} \pi O''B'' (OO''^2 + OO''BB'' + BB''^2) = \frac{2\pi}{3} \cdot 8 \cdot (1 + 5 + 25) = 519.409 \approx 519.$$

**Ответ:** 519.

# Задачи первого этапа. Информатика.

## 2.1. Первая попытка.

### Задача 2.1.1. Современная пословица (10 баллов)

В современном мире меняется всё, в том числе и наша речь: появляются новые слова, новые речевые обороты. Даже пословицы и поговорки регулярно пополняются новыми. Наш товарищ по имени Виталий зашифровал для вас пожелание в виде современной пословицы. Для этого он:

- представил каждое слово в 34-ричной системе счисления, где каждой цифрой является буква русского алфавита (без учета регистра). Каждая буква соответствует числу, равному её позиции в русском алфавите, начиная с единицы. Ни одна буква не соответствует числу 0;
- записал каждое из этих слов в десятиричной системе счисления и получил следующие три числа: 85690, 687073 и 461334847.

Расшифруйте послание Виталия и запишите в три слова, разделенных пробелом.

Примечание

привет = п · 34<sup>5</sup> + р · 34<sup>4</sup> + и · 34<sup>3</sup> + в · 34<sup>2</sup> + е · 34<sup>1</sup> + т · 34<sup>0</sup>

### Решение

Данную задачу можно решать вручную, осуществляя перевод из 10-чной системы счисления в 34-ричную. Однако, проще всего решить её при помощи программы.

### Пример программы-решения

Ниже представлена вспомогательная программа на языке Python3

```
1 letters = [chr(i) for i in range(1072, 1104)]
2 letters.insert(6, 'ё')
3 len(letters)
4
5 number = int(input())
6 word = ''
7 while(number):
8     word = letters[number % 34 - 1] + word
9     number //= 34
10
11 print(word)
```

**Ответ:** беги пока идётся.

### *Задача 2.1.2. Лес на душу (10 баллов)*

Используя открытые официальные данные Всемирного банка (международной финансовой организации, акционерами которой являются правительства 189 стран мира, — <https://data.worldbank.org/>), расставьте следующие страны по количеству покрытой лесом площади на душу их населения по состоянию на 2015 год (первыми — страны с наибольшим количеством, последними — с наименьшей).

1. Боливия
2. Габон
3. Канада
4. Монголия
5. Китай
6. Финляндия
7. Бразилия
8. Российская Федерация
9. Швеция
10. Австралия
11. Германия

### *Решение*

Для решения данной задачи требуется скачать две электронных таблицы: Forest Area (sq. km) — <https://data.worldbank.org/indicator/AG.LND.FRST.K2> и Population, total — <https://data.worldbank.org/indicator/SP.POP.TOTL>. Далее выбрать из двух таблиц колонки, соответствующие 2015 году и совместить их на одном листе, выбрать только те строки, которые соответствуют рассматриваемым странам. Затем надо вычислить отношение площади леса к количеству людей в стране и отранжировать по убыванию страны по данной колонке.

**Ответ:** 2, 3, 8, 10, 1, 4, 6, 9, 7, 5, 11.

### *Задача 2.1.3. Расхождение в ДНК (10 баллов)*

Исследователи решили проанализировать участки ДНК двух образцов. Каждый участок ДНК имеет некоторый состав нуклеотидов. Нуклеотиды ДНК образованы одним из следующих азотистых оснований: аденин, гуанин, тимин, цитозин, которые кодируются в цепочках ДНК при помощи заглавных букв по первой букве основания А, Г, Т, Ц, соответственно.

Помогите учёным определить долю расхождений в цепочках ДНК, если под расхождением понимается неравенство соответствующих нуклеотидов в цепочках, находящихся на одной позиции, считая от левого края.

## Формат входных данных

На вход программе подаётся целое число  $N$  ( $1 \leq N \leq 10^7$ ). Затем две строки в каждой из которых записана цепочка длиной  $N$  из латинских букв  $A, G, T, C$ , соответствующих азотистым основаниям.

## Формат выходных данных

Единственное число — ответ на задачу с точностью не ниже  $10^{-7}$ .

## Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

### Пример №1

<b>Стандартный ввод</b>
10 AGTCCGTCAG AGTGCCTCAG
<b>Стандартный вывод</b>
0.2

### Решение

Задача решается простым сравнением двух строк.

### Пример программы-решения

Ниже представлено решение на языке Python3

```
1 import sys
2
3 dataset = sys.stdin.read()
4
5 input = dataset.split()
6 n = int(input[0])
7 s1, s2 = input[1:]
8
9 p = 0
10
11 for i in range(n):
12     if s1[i] != s2[i]:
13         p += 1
14
15 print(p / n)
```

### Задача 2.1.4. Градусы (10 баллов)

Для различных картографических сервисов требуются различные форматы координат. Чаще всего координаты обозначаются в десятичных градусах (DD – decimal degrees), например, так: (55.7644871, 37.6602897)

А для некоторых сервисов требуется перевод координат в формат  $(A_1^\circ B_1' C_1'' D_1, A_2^\circ B_2' C_2'' D_2)$ , где  $A$  – градусы,  $B$  – минуты,  $C$  – секунды, и направление  $D$  ( $N$  – северная широта,  $S$  – южная широта,  $W$  – западная долгота,  $E$  – восточная долгота).

Напишите программу-калькулятор, осуществляющую вышеописанный перевод координат.

#### Формат входных данных

На вход программе подается строка, в которой записаны два вещественных числа с 7-ю знаками после точки: широта  $X$  ( $-90.0000000 \leq X \leq 90.0000000$ ) и долгота  $Y$  ( $-180.0000000 \leq Y \leq 180.0000000$ ) некоторого объекта на карте.

#### Формат выходных данных

Выведите в отдельной строке координаты в требуемом формате. Обратите внимание, что значения  $A, B, C$  в формате указываются с ведущими нулями до двух знаков. Округление осуществляйте по модулю в меньшую сторону.

#### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

<b>Стандартный ввод</b>
55.7644871 37.6602897
<b>Стандартный вывод</b>
55°45'52'' N 37°39'37'' E

#### Пример №2

<b>Стандартный ввод</b>
0.0000000 0.0000000
<b>Стандартный вывод</b>
00°00'00'' N 00°00'00'' E

## Решение

Особенность данной задачи в целочисленной арифметике. Если оперировать с входными данными как с вещественными числами, то некоторые числа, не представимые конечным числом знаков в двоичной системе счисления, переводились неверно. Например, правильный результат для теста  $(23.2000000 - 34.8000000) \rightarrow (23^\circ 12'00'' N 34^\circ 48'00'' W)$ , а не  $23^\circ 11'59'' N 34^\circ 47'59'' W$ .

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1  import sys
2
3  dataset = sys.stdin.read()
4
5  w1, w2 = dataset.split()
6  w1 = w1.split('.')
7  w2 = w2.split('.')
8
9  a1 = int(''.join(w for w in w1))
10 a2 = int(''.join(w for w in w2))
11
12 d1 = 'N'
13 d2 = 'E'
14 if a1 < 0:
15     d1 = 'S'
16     a1 *= -1
17 if a2 < 0:
18     d2 = 'W'
19     a2 *= -1
20
21 x1 = a1 % 10000000
22 x2 = a2 % 10000000
23
24 a1 //= 10000000
25 a2 //= 10000000
26
27 x1 = x1 * 3600 // 10000000
28 b1 = x1 // 60
29 c1 = x1 % 60
30
31 x2 = x2 * 3600 // 10000000
32 b2 = x2 // 60
33 c2 = x2 % 60
34
35 ans = "{:02d}°{:02d}'{:02d}''{} {}{:02d}°{:02d}'{:02d}''{}\n".
36     format(a1, b1, c1, d1, a2, b2, c2, d2)
37
38 print(ans)

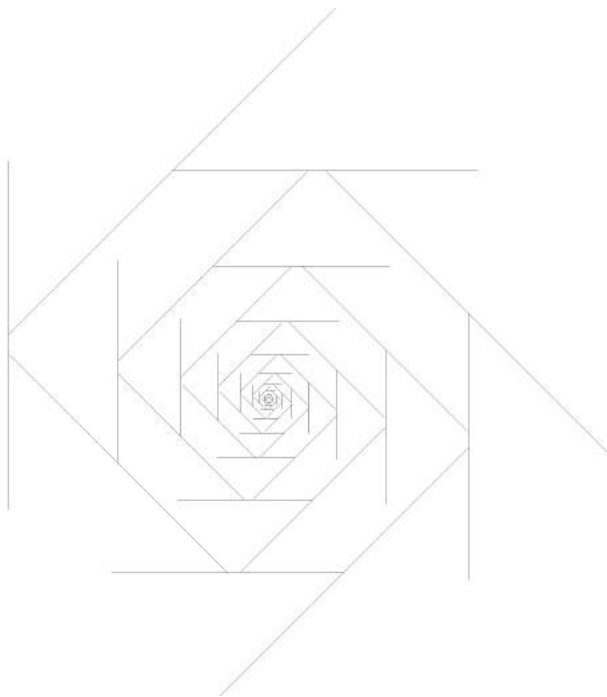
```

### Задача 2.1.5. Фрактал (10 баллов)

На сегодняшний день одним из самых быстро развивающихся и перспективных видов компьютерной графики является фрактальная графика. На этот раз мы и тебе предлагаем немного порисовать, но в достаточно простой геометрии. Основным повторяющимся элементом нашего рисунка будет отрезок длиной  $n$  пикселей.



- Первый элемент располагается вертикально, считаем, что нижний конец его закреплен, а верхний свободен. Его длина  $s_1 = n$ .
- Каждый последующий элемент имеет меньшую длину, которая определяется по формуле  $s_{i+1} = a \cdot s_i // b$ , где знак  $//$  обозначает целочисленное деление. Если  $s_i = 0$ , элемент не отображается.
- Для каждого элемента определяется точка крепления нового элемента. Этот элемент мы делим в соотношении  $c/d$  таким образом, чтобы отрезок у свободного конца был короче, чем у закрепленного. Местом прикрепления выбирается пиксель, на который попала точка разбиения элементов на отрезки. Если местоположение пикселя не определяется точно (точка достижения соотношения находится между двумя пикселями), то положение места прикрепления определяется из двух пикселей по принципу близости к свободному концу.
- Новый элемент прикрепляется под углом  $45^\circ$  к наиболее короткой части по часовой стрелке от предыдущего. Для параметров  $n = 1000, a = 935, b = 1000, c = 1, d = 1$  фрактал будет иметь вид:



Сколько будет закрасненных пикселей на рисунке?

### Формат входных данных

Строка, в которой указаны целочисленные параметры  $n, a, b, c, d$ , разделенные пробелом, где  $(10^{10000} \leq n \leq 10^{10001}, 1 \leq a, b \leq 1000, 0 < a/b \leq 0.935, 1 \leq c, d \leq 10)$

### Формат выходных данных

Единственное целое число.

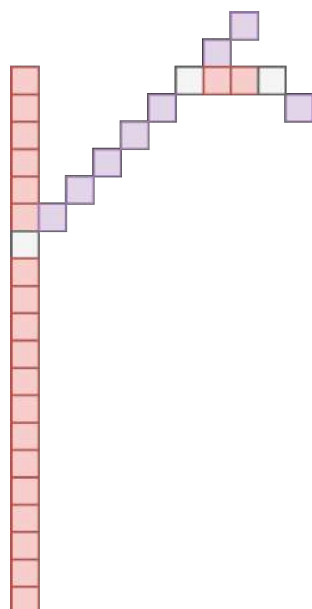
*Пример №1*

<b>Стандартный ввод</b>
20 2 5 1 2
<b>Стандартный вывод</b>
32

### Пояснения к ответу

- Первый элемент длиной 20. Определяем длину следующего элемента:  $20 \cdot 2//5 = 8$ . Определяем точку крепления нового элемента:  $20 \cdot 1/(1+2) = 6\frac{2}{3}$  — попадает на 7-й от свободного края пиксель. Крепим элемент к первому элементу под углом  $45^\circ$  к короткой части по направлению по часовой стрелке.
- Второй элемент 8. Определяем длину следующего элемента:  $8 \cdot 2//5 = 3$ . Определяем точку крепления нового элемента:  $8 \cdot 1/(1+2) = 2\frac{2}{3}$  — попадает на 3-й от свободного края пиксель. Крепим его ко второму элементу под углом  $45^\circ$  к короткой части по направлению по часовой стрелке.
- Третий элемент 3. Определяем длину следующего элемента:  $3 \cdot 2//5 = 1$ . Определяем точку крепления нового элемента:  $3 \cdot 1/(1+2) = 1$  — попадает на стык, поэтому выбираем пиксель ближе к свободному краю. Крепим его к третьему элементу под углом  $45^\circ$  к короткой части по направлению по часовой стрелке.
- Четвертый элемент 1. Определяем длину следующего элемента:  $1 \cdot 2//5 = 0$ . Значит, этот элемент последний.

На иллюстрации красным и сиреневым обозначены элементы фрактала, серым точки, крепления.



### Решение

Из анализа фрактала можно сделать вывод, что линии, из которых он состоит, не пересекаются. Таким образом, на основании предыдущей длины путем арифмети-

ческих операций (длинная арифметика) можно вычислить длину нового фрагмента. В качестве ответа вводится одно длинное число.

### Пример программы-решения

Ниже представлена вспомогательная программа на языке Python3

```
1 import sys
2
3 dataset = sys.stdin.read()
4 n, a, b, c, d = map(int, dataset.split())
5 ans = 0
6 while n:
7     ans += n
8     n = n * a // b
9
10 print(ans)
```

### Задача 2.1.6. Система передатчиков (10 баллов)

Инженер разрабатывает оптимальный маршрут для передачи данных через систему передатчиков. Все передатчики располагаются на прямоугольной сетке. Каждый из передатчиков может передавать сообщение только ближайшим соседям справа и снизу, если они есть. Определите наибольшую пропускную способность от левого верхнего передатчика до правого нижнего, если Вам известны пропускные способности каждого из передатчиков.

#### Формат входных данных

На вход программе подаются размеры прямоугольника: длина  $n$  и ширина  $m$  ( $1 \leq n \leq 1000, 1 \leq m \leq 1000$ ).

В следующих  $n$  строках записываются  $m$  целых чисел – максимальная пропускная способность каждого из передатчиков  $k$  ( $0 \leq k \leq 1000000$ ).

#### Формат выходных данных

Выведите целое число – максимально возможную пропускную способность имеющейся системы.

#### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

*Пример №1*

Стандартный ввод
5 6
8 7 6 5 0 0
4 3 1 8 1 8
4 4 4 5 0 4
7 8 9 3 6 0
1 4 2 2 4 6
Стандартный вывод
3

### Решение

Данная задача решается алгоритмом динамического программирования. Для каждого передатчика определяется наиболее оптимальный сосед, благодаря которому может быть достигнута наибольшая пропускная способность.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4 input = dataset.split()
5 n = int(input[0])
6 m = int(input[1])
7 matrix = [[int(input[2 + x * m + y]) for y in range(m)] for x in range(n)]
8
9 for i in range(n):
10     for j in range(m):
11         list = []
12         if i > 0:
13             list.append(matrix[i - 1][j])
14         if j > 0:
15             list.append(matrix[i][j - 1])
16         if list:
17             matrix[i][j] = min(matrix[i][j], max(list))
18
19 print(matrix[n - 1][m - 1])

```

### Задача 2.1.7. Посади деревце (10 баллов)

Инженер-программист решил соорудить ограждение для деревца. Для этого он включил генератор случайных чисел и получил координаты точек, в которые он вобьет колышки и протянет между ними веревку. Сможет ли наш герой поместить в это сооружение деревце, чтобы оно лежало внутри ограждения? Для посадки деревца достаточно любой ненулевой площади.

### Формат входных данных

На вход программе в одной строке подаются координаты трёх точек  $a_x, a_y, b_x, b_y, c_x, c_y$ . Значения абсцисс и ординат целочисленные и не превышают по модулю  $10^9$ .

## Формат выходных данных

Проверьте, может ли инженер посадить в описанном ограждении дерево. Выведите "Yes" при положительном ответе или "No" в противном случае.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

<b>Стандартный ввод</b>
10 0 0 10 10 10
<b>Стандартный вывод</b>
Yes

#### Решение

В данной задаче требуется проверить, принадлежность точек одной прямой. Точки  $(x_1, y_1)$ ,  $(x_2, y_2)$ ,  $(x_3, y_3)$  лежат на одной прямой, если:

$$\frac{y_2 - y_1}{x_2 - x_1} = \frac{y_3 - y_1}{x_3 - x_1}.$$

Так как работа с вещественными числами всегда нежелательно, то избавимся в данной проверке от деления:

$$(y_2 - y_1) \cdot (x_3 - x_1) = (x_2 - x_1) \cdot (y_3 - y_1).$$

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 def is_in_line(x1, y1, x2, y2, x3, y3):
4     return (x3 - x1) * (y2 - y1) == (y3 - y1) * (x2 - x1)
5
6 dataset = sys.stdin.read()
7 x1, y1, x2, y2, x3, y3 = [int(x) for x in dataset.split()]
8 if is_in_line(x1, y1, x2, y2, x3, y3):
9     print("No")
10 print("Yes")

```

### Задача 2.1.8. Полиномиальный характер (10 баллов)

В результате некоторого эксперимента исследователь получил график. Он обратил внимание, что график описывает некоторую непрерывную функцию, которая

пересекает ось абсцисс в некотором количестве точек. Ещё он отметил, что нет таких участков, в которых график касается оси абсцисс, не пересекая её. А значения функции на левом конце графика всегда отрицательны.

У исследователя возникло предположение, что данная функция полиномиальная. И для того, чтобы это проверить, он решил записать все значения, в которых функция пересекла ось абсцисс и построить по данным значениям полиномиальную функцию, а дальше наложить графики и оценить расхождение. Помогите исследователю составить полиномиальную функцию, проходящую через заданные точки. При этом порядок данной функции не должен быть выше количества точек, а коэффициент перед старшей степенью должен быть по модулю равен 1.

### Формат входных данных

Программа в первой строке принимает на вход количество точек  $n$  ( $1 \leq n \leq 10$ ).

Далее в следующей строке вводятся  $n$  аргументов  $x_i$  ( $-100 \leq x_i \leq 100$ ;  $x_i \in \mathbb{Z}; \forall i, j : x_i \neq x_j$ ), в которых функция принимает нулевое значение.

### Формат выходных данных

В качестве ответа выведите в отдельной строке через пробел  $n + 1$  целочисленных коэффициентов полинома, начиная со старшей степени.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

### Пример №1

<b>Стандартный ввод</b>
2 -5 3
<b>Стандартный вывод</b>
-1 -2 15

### Решение

Если полиномиальная функция проходит через точки  $x_0, x_1, \dots, x_n$ , то полиномиальная функция с наименьшим порядком будет равна по модулю:

$$|f(x)| = (x - x_0)(x - x_1) \cdot \dots \cdot (x - x_n).$$

Если значение функции в левом конце отрицательно, то если функция, проходит через чётное количество точек,

$$f(x) = -(x - x_0)(x - x_1) \cdot \dots \cdot (x - x_n).$$

Далее задача сводится к перемножению полиномов 1-й степени.

## Пример программы-решения

Ниже представлено решение на языке Python3

```
1 import sys
2
3 dataset = sys.stdin.read()
4 lst = list(map(int, dataset.split()))
5 n = lst[0]
6 lst = lst[1:]
7 ans = [0] * (n + 1)
8 ans[0] = 1
9
10 for i in range(n):
11     for j in range(i + 1, 0, -1):
12         ans[j] -= ans[j - 1] * lst[i]
13
14 if n % 2 == 0:
15     ans = [-x for x in ans]
16
17 print(' '.join(str(x) for x in ans))
```

### *Задача 2.1.9. Длина пути (10 баллов)*

Робот умеет двигаться по траектории, задаваемой некоторой полиномиальной функцией  $f(x)$  с целочисленными показателями. Свой путь он начинает в координате  $(0, f(0))$  и двигается в сторону возрастания значений по оси абсцисс. Но, как вы знаете, на любое движение нужна энергия, наш робот не исключение. Он снабжен аккумулятором, которые позволяет ему проехать путь длиной  $s$ . Где окажется робот в конце пути?

#### Формат входных данных

В первой строке задается  $n$  ( $0 \leq n \leq 5$ ) — степень полинома, задающего траекторию робота.

Во второй строке через пробел задаются целочисленные значения  $a_i$  ( $-10 \leq a_i \leq 10$ ) — коэффициенты перед степенями многочлена, в порядке от старшей степени к младшей.

В третьей строке вещественное число  $s$  ( $0 \leq s \leq 100$ ) — длина пути робота.

#### Формат выходных данных

В отдельной строке абсцисса координаты, где остановится робот, с абсолютной точностью не менее  $10^{-3}$ .

#### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

*Пример №1*

Стандартный ввод
0
5
100
Стандартный вывод
100.00000002054185

*Пример №2*

Стандартный ввод
1
1 1
100
Стандартный вывод
70.71068001124502

*Пример №3*

Стандартный ввод
2
2 3 5
100
Стандартный вывод
6.350909999928737

*Решение*

Для решения задачи требуется разделить интервал по  $x$  на мельчайшие части и определить расстояние между последовательно идущими точками в порядке возрастания абсцисс. Таким образом будет получено приближенное значение длины кривой.

**Пример программы-решения**

Ниже представлено решение на языке Python3

```

1 import sys
2
3 def f(x, n, coeffs):
4     res = 0
5     for i in range(n + 1):
6         res *= x
7         res += coeffs[i]
8     return res
9
10 def way(x0, step, n, coeffs):
11     return math.sqrt(step ** 2 + (f(x0 + step, n, coeffs) - f(x0, n, coeffs)) ** 2)
12
13 dataset = sys.stdin.read()
14 lst = dataset.split()

```



```

15
16 n = int(lst[0])
17 s = float(lst[-1])
18 coeffs = list(map(int, lst[1:-1]))
19 wayval = 0
20 x = 0.0
21 step = 0.00001
22
23 while(wayval < s):
24     wayval += way(x, step, n, coeffs)
25     x += step
26
27 print(str(x))

```

### **Задача 2.1.10. Исчезновение леса (10 баллов)**

Обезлесение является актуальной проблемой во многих частях земного шара, поскольку влияет на экологические, климатические и социально-экономические характеристики и снижает качество жизни. Довольно часто лес исчезает участками. Благодаря космическим снимкам мы можем определять, где лес растет, а где его нет. Каждый снимок с космоса проходит сложную предобработку.

Для упрощения анализа, мы предлагаем рассмотреть снимки в следующем предобработанном формате: пометим каждый пиксель на котором изображен лес символом '.', а тот, на котором нет леса — 'x'.

Ваша задача будет заключаться в том, чтобы посчитать количество участков, в которых лес отсутствует. Под участком будем понимать набор таких пикселей без леса, которые касаются минимум в одной точке хотя бы с ещё одним пикселем данного набора. Или не касаются ни одного пикселя без леса.

Данную задачу можно было бы решать вручную, однако сколько это займет по времени, если рассматривать мелкие-мелкие пиксели на сотнях или даже тысячах изображений? Вы однозначно умеете делать это быстрее!

#### **Формат входных данных**

В первой строке задается целое число  $t$  ( $1 \leq t \leq 100$ ) — количество тестов.

Далее с новой строки для каждого теста задается целое число  $n$  ( $1 \leq n \leq 100$ ) — размер стороны квадратного изображения в пикселях.

И в следующих  $n$  строках указывается строка из пикселей ('x' и '.') длиной  $n$ .

#### **Формат выходных данных**

В отдельной строке  $t$  целых чисел через пробел — количество горящих участков для каждого из изображений.

#### **Система оценки**

Примерно в 40% тестов  $50 \leq t \leq 100$ . Примерно в 75% тестов  $50 \leq n \leq 100$ .

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.



```

1 import sys
2
3 def fire_clean(picture, i, j):
4     visited = [[0 for _ in range(len(picture))] for _ in range(len(picture))]
5     nodes = [(i, j)]
6
7     while nodes:
8         i, j = nodes[0]
9         nodes = nodes[1:]
10        if i < 0 or i >= len(picture) or j < 0 or j >= len(picture):
11            continue
12        if visited[i][j]:
13            continue
14        visited[i][j] = 1
15        if picture[i][j] == 'x':
16            picture[i][j] = '.'
17            nodes += [
18                (i - 1, j - 1),
19                (i - 1, j),
20                (i - 1, j + 1),
21                (i, j - 1),
22                (i, j + 1),
23                (i + 1, j - 1),
24                (i + 1, j),
25                (i + 1, j + 1),
26            ]
27        return
28
29 def solve_test(n, picture):
30     groups = 0
31     for i in range(n):
32         for j in range(n):
33             if picture[i][j] == 'x':
34                 groups += 1
35                 fire_clean(picture, i, j)
36     return str(groups) + ' '
37
38
39 dataset = sys.stdin.read()
40 d = dataset.split()
41 tn = int(d[0])
42 idx = 1
43 eprint(tn)
44 ans = ''
45 for _ in range(tn):
46     ans += solve_test(int(d[idx]), [list(x) for x in d[idx + 1:idx + 1 + int(d[idx])]])
47
48     idx += int(d[idx]) + 1
49
50 print(ans)

```

## 2.2. Вторая попытка.

### Задача 2.2.1. Queen (10 баллов)

Валера имеет плохую память на номера телефонов. Особенно тяжело ему пришлось с номерами мобильных телефонов, у которых всегда приходится запоминать

помимо 7 цифр номера, еще две, определяющие код оператора ( $8(9^{**})_{-}^{***}_{-}^{**}_{-}^{**}$ ). Про телефонную книгу в телефоне он знал, но запоминание было принципом.

Узнав про системы счисления он возрадовался тому, что числа можно запоминать при помощи букв, да и при большом основании они будут ощутимо короче. Благодаря родителям он очень полюбил группу Queen, и когда пришла пора дарить первый телефон младшей сестренке, Валера выбрал ей номер так, что в некоторой системе счисления число из 9 цифр для запоминания было равно QUEEN и основание системы было наибольшее из возможных для данного формата номера телефона. Напоминаем, что при нехватке цифр для записи разрядов используются буквы латинского алфавита. Запишите в ответе те самые 9 цифр.

### *Решение*

Проще всего данную задачу решать методом программного перебора. Очевидно, что основание системы счисления не превышает 100, так как при помощи 5 цифр некоторой системы счисления кодируется девятизначное число в десятичной системе счисления.

### Пример программы-решения

Ниже представлена вспомогательная программа на языке Python3

```

1 letters = [chr(i) for i in range(1072, 1104)]
2 letters.insert(6, 'ë')
3 len(letters)
4
5 number = int(input())
6 word = ''
7 while(number):
8     word = letters[number % 34 - 1] + word
9     number //= 34
10
11 print(word)

```

Ответ: 976714307.

### *Задача 2.2.2. Равенство полов (10 баллов)*

Используя открытые официальные данные Всемирного банка (международной финансовой организации, акционерами которой являются правительства 189 стран мира, — <https://data.worldbank.org/>), укажите название той страны, в которой наименьшая численная разница между мужчинами и женщинами по данным за 2017 год. Укажите полное официальное русское название этой страны.

### *Решение*

Для решения данной задачи необходимо скачать две электронных таблицы: Population, female — <https://data.worldbank.org/indicator/SP.POP.TOTL.FE.IN> и Population, male — <https://data.worldbank.org/indicator/SP.POP.TOTL.MA.IN>. Из

каждой таблицы необходимо перенести на отдельный лист данные за 2017 год, затем найти страну, модуль разности по численности между мужчинами и женщинами минимален.

**Ответ:** Республика Македония.

### **Задача 2.2.3. Little to Big (10 баллов)**

В современной вычислительной технике и цифровых системах связи информация обычно представлена в виде последовательности байтов. Если число не может быть представлено одним байтом, имеет значение, в каком порядке байты записываются в памяти компьютера или передаются по линиям связи. Часто выбор порядка записи байтов произволен и определяется только соглашениями.

В общем случае, для представления числа  $M$ , большего 255, приходится использовать несколько байтов. При этом число  $M$  записывается в позиционной системе счисления с основанием 256:

$$M = \sum_{i=0}^n A_i \cdot 256^i = A_0 \cdot 256^0 + A_1 \cdot 256^1 + A_2 \cdot 256^2 + \dots + A_n \cdot 256^n.$$

Набор целых чисел  $A_0, \dots, A_n$ , каждое из которых лежит в интервале от 0 до 255, является последовательностью байтов, составляющих  $M$ . При этом  $A_0$  – младший байт,  $A_n$  – старший байт числа  $M$ .

Есть несколько способов записи целых чисел:

1. Big-Endian – запись числа от старшего байта к младшему,
2. Little-Endian – запись числа от младшего байта к старшему.

Ваш компьютер поддерживает big-endian, а требуется работать с устройством, которое посылает 32-битные беззнаковые целые числа в формате little-endian. Напишите программу, которая адаптирует формат получаемого числа для работы на компьютере.

#### **Формат входных данных**

В единственной строке целое неотрицательное 32-битное число.

#### **Формат выходных данных**

Единственное число – десятичная запись числа в формате big-endian, полученная путем перевода этого числа из little-endian.

#### **Система оценки**

Баллы за задачу будут начислены, если все тесты будут пройдены успешно

#### *Пример №1*

<b>Стандартный ввод</b>
3496683923
<b>Стандартный вывод</b>
2468965328

## Решение

В этой задаче важно обратить внимание на то, что длина подаваемого на вход числа равна 32 битам, то есть 4 байтам. То есть, например, для числа 255 ответ 4278190080, а не 255.

## Пример программы-решения

Ниже представлено решение на языке Python3

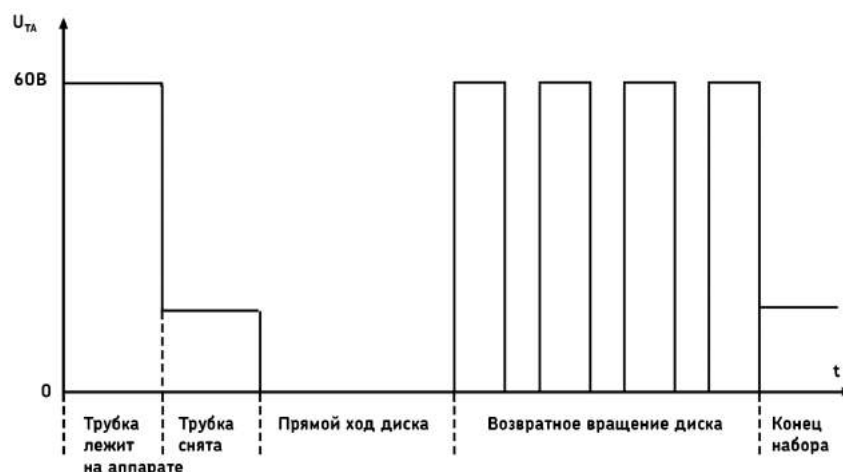
```
1 import sys
2
3 dataset = sys.stdin.read()
4
5 n = int(dataset)
6 ans = 0
7
8 for _ in range(4):
9     ans *= 256
10    ans += n % 256
11    n //= 256
12
13 print(ans)
```

## Задача 2.2.4. Размыкания (10 баллов)

Буквально каждый из нас пользуется мобильным телефоном. Некоторые из нас не могут себе представить популярное устройство связи в будущем. А знаете ли вы, что лет эдак 40 назад никто не мог предположить о том, что телефоны станут не просто кнопочными, но и мобильными, да ещё и с сенсорным экраном. Раньше многие телефоны выглядели следующим образом:



Для такого телефонного аппарата (с дисковым номеронабирателем) набор номера абонента осуществляется следующим образом: при вращении диска по часовой стрелке до пальцевого упора контакты номеронабирателя замыкают линию, а при возвратном вращении линия размыкается такое число раз, которое соответствует набранной цифре. На рисунке показана временная диаграмма работы телефонного аппарата.



При наборе номера 8-800-000-00-00 в сумме будет произведено 106 размыканий (2 по 8 и 9 по 10). Посчитайте, сколько существует возможных различных номеров (11-значных комбинаций цифр, в том числе и с ведущими нулями), при которых возникает заданное количество размыканий  $n$ .

### Формат входных данных

В отдельной строке целое число  $n$  ( $0 \leq n \leq 1000$ )

### Формат выходных данных

Единственное число — ответ на задачу.

### Решение

В условиях ограниченного времени (5 минут) правильнее всего написать программу, вычисляющую ответ.

### Пример программы-решения

Ниже представлена вспомогательная программа на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4
5 n = int(dataset.split()[0])
6
7 d = {0:1}
8
9 for _ in range(11):
10     tmp = {}
11     for x in d.keys():
12         for digit in range(10):
13             if ((x + digit + 1) in tmp.keys()):
14                 tmp[x + digit + 1] += d[x]
15     else:
```

```
16         tmp[x + digit + 1] = d[x]
17     d = tmp.copy()
18
19     if (n in d.keys()):
20         ans = d[n]
21     else:
22         ans = 0
23
24     print(ans)
```

### **Задача 2.2.5. Статистика по салатам (10 баллов)**

В столовой учащиеся информационно-технологического класса решили провести анализ того, чем их кормят. Для исследований они выбрали овощной салат из капусты, горошка, кукурузы и фасоли, который повара столовой готовят достаточно часто. Ребята в каждой порции считали количество горошин, зернышек кукурузы и фасолин. Через некоторое время ребята ощутили маленький размер выборки и вовлекли в этот процесс большое количество учащихся школ, которых обслуживает тот же комбинат питания, что и их школу. Так за учебный год у ребят набралась неплохая выборка. Для заданной выборки определите, сколько в среднем ингредиентов каждого типа (горох, кукуруза и фасоль) встречаются в салате и каково медианное значение данных ингредиентов в салате. После этого для каждого типа ингредиентов отбросьте 10% наименьших значений и 10% наибольших, затем снова определите среднее значение и медиану.

#### **Пояснения к ответу**

Если 10% от количества элементов является вещественным числом, округлите его до целого в меньшую сторону. Для выборки из четного количества элементов медиана вычисляется как полусумма двух соседних значений в середине диапазона.

#### **Формат входных данных**

В первой строке подается целое значение  $N (1 \leq N \leq 10^6)$  – количество исследованных салатов.

Далее в  $N$  строках через пробел подаются три целых числа  $a_i$ ,  $b_i$  и  $c_i$  ( $0 \leq a_i, b_i, c_i \leq 100$ ) – количество единиц горошка, кукурузы и фасоли в  $i$ -м салате соответственно.

#### **Формат выходных данных**

В четырёх строках выведите по 3 числа – требуемые параметры для гороха, кукурузы и фасоли соответственно, с точностью не ниже  $10^{-6}$ :

В первой строке - среднее значение каждого из ингредиентов в полной выборке.

Во второй строке - медианное значение каждого из ингредиентов в полной выборке.

В третьей строке - среднее значение каждого из ингредиентов в усеченной выборке.



В четвертой строке - медианное значение каждого из ингредиентов в усеченной выборке.

### Система оценки

Примерно в 50% тестов  $10^5 \leq N \leq 10^6$ .

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

Стандартный ввод		
3		
91 6 34		
64 31 16		
77 5 92		
Стандартный вывод		
77.33333333333333	14.0	47.33333333333336
77.0	6.0	34.0
77.33333333333333	14.0	47.33333333333336
77.0	6.0	34.0

#### Пример №2

Стандартный ввод		
12		
81 36 38		
17 89 30		
98 70 20		
74 68 20		
78 43 1		
0 17 31		
41 25 34		
14 12 55		
70 23 39		
43 84 83		
93 17 97		
38 91 25		
Стандартный вывод		
53.91666666666664	47.91666666666664	39.41666666666664
56.5	39.5	32.5
54.9	47.2	37.5
56.5	39.5	32.5

#### Решение

Медиана - это такое число выборки, что половина из элементов выборки больше него или равна ему, а другая половина меньше него или равна ему. Для выборки из четного количества элементов медиана вычисляется как полусумма двух соседних

значений в середине упорядоченной последовательности элементов выборки. Задача заключается в том, чтобы найти среднее (которое изменяется при усечении выборки) и найти медиану (которая не изменится при усечении выборки).

### Пример программы-решения

Ниже представлено решение на языке Python3

```
1 import sys
2
3 def get_med(q, n):
4     ans = [0, 0, 0]
5     size = n
6
7     for i in range(3):
8         pos = 0
9         idx = 0
10        val1 = (size - 1) // 2
11        val2 = (size - 0) // 2
12        #eprint(q[i])
13        while (pos + q[i][idx] <= val1):
14            #eprint(pos, val1, idx)
15            pos += q[i][idx]
16            idx += 1
17        val1 = idx
18
19        while (pos + q[i][idx] <= val2):
20            pos += q[i][idx]
21            idx += 1
22        val2 = idx
23
24        ans[i] = (val1 + val2) / 2
25
26    return ans
27
28 dataset = sys.stdin.read()
29
30 d = dataset.split()
31 n = int(d[0])
32 numbers = list(map(int, d[1:]))
33
34 q = []
35
36 q.append([0] * 101)
37 q.append([0] * 101)
38 q.append([0] * 101)
39
40 sum1 = [0, 0, 0]
41
42 for i in range(3 * n):
43     q[i % 3][numbers[i]] += 1
44
45 sum1 = [sum([q[j][i] * i for i in range(101)]) for j in range(3)]
46
47 med = get_med(q, n)
48
49 for i in range(3):
50     ten_p = n // 10
51     idx = 0
```

```

52     while (ten_p):
53         x = min(q[i][idx], ten_p)
54         ten_p -= x
55         q[i][idx] -= x
56         idx += 1
57
58 for i in range(3):
59     ten_p = n // 10
60     idx = 100
61     while (ten_p):
62         x = min(q[i][idx], ten_p)
63         ten_p -= x
64         q[i][idx] -= x
65         idx -= 1
66
67 sum2 = [sum([q[j][i] * i for i in range(101)]) for j in range(3)]
68
69 print(' '.join(str(x / n) for x in sum1))
70 print(' '.join(str(x) for x in med))
71 print(' '.join(str(x / (n - n // 10 - n // 10)) for x in sum2))
72 print(' '.join(str(x) for x in med))

```

### **Задача 2.2.6. "Репликация" (10 баллов)**

Василий решил, что создание своей собственной системы управления базами данных (СУБД) — посильная для него задача, и даже немного подумал про отказоустойчивость. Он продумал собственную систему "репликации" (резервного копирования) данных, чтобы быть уверенным в их сохранности. Он активно использовал её для хранения некоторых своих измерений. Однажды алгоритм "репликации" дал сбой и Василий определил, что одна из записей в "реплику" была занесена дважды. Василий выгрузил в текстовый формат колонку с числовыми значениями идентификаторов записей, которые были представлены 32-битными целыми числами. Одна беда — они выгрузились в хаотичном порядке. Теперь у Василия два текстовых файла с числами: из базы и из реплики. Помогите как можно быстрее найти идентификатор дважды повторенной записи.

#### **Формат входных данных**

В первой строке  $n$  ( $1 \leq n \leq 10^6$ ) — число записей в базе Василия.

Во второй строке  $n$  различных целых неотрицательных 32-битных значений  $a_i$ , разделенных пробелом — идентификаторы записей в базе Василия.

В третьей строке  $n + 1$  целых неотрицательных 32-битных значений  $b_i$ , разделенных пробелом — идентификаторы записей в "реплике" Василия.

#### **Формат выходных данных**

Единственное число — ответ на задачу

#### **Система оценки**

Примерно в 50% тестов  $10^5 \leq n \leq 10^6$ .

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

### Пример №1

<b>Стандартный ввод</b>
8 22 23 11 8 21 2 6 18 18 21 2 11 23 6 23 22 8
<b>Стандартный вывод</b>
23

### Решение

Так как значения в базе целочисленные и хранятся в двоичном виде, то можно воспользоваться логической операцией XOR. Её особенность в том, что

$$\forall x : x \text{ XOR } x = 0, x \text{ XOR } y = y \text{ XOR } x.$$

Таким образом, например,

$$x \text{ XOR } y \text{ XOR } x \text{ XOR } z \text{ XOR } y = z.$$

Следовательно, если мы совершим эту операцию для всех считанных целых чисел (кроме количества в первой строке), то получим единственное значение, которое не повторялось.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4
5 d = list(map(int, dataset.split()))
6
7 ans = 0
8
9 for i in range(1, d[0] * 2 + 2):
10     ans = ans ^ d[i]
11
12 print(ans)

```

### Задача 2.2.7. Параллелограмм (10 баллов)

Игорь решил заняться необычной графикой. Сначала он по трём точкам на плоскости рисует треугольники. Затем ищет красивый способ дорисовать треугольник до параллелограмма. Красивым способом он считает такой, при котором:

1. У параллелограмма одна из диагоналей получается наидлиннейшей из возможных вариантов.
2. Если таких вариантов построения несколько, Игорь из них выбирает тот, у которого сумма координат новой точки (абсциссы + ординаты) наибольшая.
3. Ну а если и таких точек несколько, то из них он выбирает ту, у которой которой наибольшая абсцисса.

По трем заданным точкам треугольника укажите координаты четвертой точки получаемого красивого параллелограмма.

### Формат входных данных

В трех строчках через пробел по паре целых чисел — координаты вершин треугольника.

$$x_i, y_i (-10^7 \leq x_i, y_i \leq 10^7)$$

Гарантируется, что они не лежат на одной прямой.

### Формат выходных данных

Абсцисса и ордината точки — ответ на задачу

### Система оценки

Примерно в 50% тестов  $-10^3 \leq x_i, y_i \leq 10^3$ .

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

### Пример №1

<b>Стандартный ввод</b>
10 0 0 10 10 10
<b>Стандартный вывод</b>
20 0

### Решение

Сначала надо определить максимальную длину наибольшей возможной диагонали. Далее определяем список вершин тех параллелограммов, у которых хотя бы одна из диагоналей является максимальной.

Треугольников с максимальной диагональю может быть не более 2, так как построить равносторонний треугольник с целыми координатами нельзя. Поэтому если есть два треугольника с максимальными диагоналями выбираем из них по дополнительным критериям.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 def dist(x1, y1, x2, y2):
4     return (x2 - x1) ** 2 + (y2 - y1) ** 2
5
6 def max_diag(l):
7     a1 = (l[4] - l[0]) ** 2 + (l[5] - l[1]) ** 2
8     a2 = (l[6] - l[2]) ** 2 + (l[7] - l[3]) ** 2
9     return max(a1, a2)
10
11 dataset = sys.stdin.read()
12
13 p = [int(a) for a in dataset.split()]
14 q = [0] * 3
15
16 q[0] = p[:]
17 q[1] = p[:]
18 q[2] = p[:]
19
20 v = [[p[(2 * x + 2) % 6] + p[(2 * x + 4) % 6] - p[(2 * x + 6) % 6],
21       p[(2 * x + 3) % 6] + p[(2 * x + 5) % 6] - p[(2 * x + 1) % 6]] for x in range(3)]
22
23 max_value = 0
24 for i in range(3):
25     q[i] = q[i][: ((i + 2) * 2) % 6] + v[i] + q[i][((i + 2) * 2) % 6:]
26     max_value = max(max_value, max_diag(q[i]))
27
28 ansp = []
29
30 for i in range(3):
31     if max_value == max_diag(q[i]):
32         ansp.append(v[i])
33
34 ans = []
35 max_sum = 0
36
37 if len(ansp) == 2:
38     if sum(ansp[0]) > sum(ansp[1]):
39         ans = ansp[0]
40     elif sum(ansp[0]) < sum(ansp[1]):
41         ans = ansp[1]
42     elif ansp[0][0] > ansp[1][0]:
43         ans = ansp[0]
44     else:
45         ans = ansp[1]
46 else:
47     ans = ansp[0]
48
49 print('{} {}'.format(ans[0], ans[1]))

```

### *Задача 2.2.8. Движение по шару (10 баллов)*

Люди перемещаются по поверхности нашей планеты. Они ходят, бегают, ездят и даже летают. И им всегда важно расстояние до объекта, пусть даже они почти никогда не двигаются по прямой. По координатам двух объектов в десятичных градусах определите расстояние между ними на поверхности Земли, если считаем её сферой

с радиусом  $R = 6371302$  м.

### Формат входных данных

В единственной строке через пробел подаются вещественные значения  $lat_1, long_1, lat_2, long_2$  ( $-90 \leq lat_i \leq 90, -180 \leq long_i \leq 180$ ) — координаты двух объектов.

### Формат выходных данных

Единственное целое число — ответ на задачу в километрах. Округление дробной части производите к ближайшему целому.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

<b>Стандартный ввод</b>
55 37.620393 43.116418 131.882475
<b>Стандартный вывод</b>
6458

### Решение

Для решения данной задачи с указанной точностью достаточно вывести или найти в дополнительных источниках формулу нахождения расстояния между двумя точками на поверхности шара и запрограммировать вычисления.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2 import math
3
4 dataset = sys.stdin.read()
5 x1, y1, x2, y2 = map(float, dataset.split())
6 rad = 6371.302
7
8 lat1 = x1 * math.pi / 180.
9 lat2 = x2 * math.pi / 180.
10 long1 = y1 * math.pi / 180.
11 long2 = y2 * math.pi / 180.
12
13 c11 = math.cos(lat1)
14 c12 = math.cos(lat2)
15 s11 = math.sin(lat1)
16 s12 = math.sin(lat2)

```

```
17
18 delta = long2 - long1
19 cdelta = math.cos(delta)
20 sdelta = math.sin(delta)
21
22 y = math.sqrt(math.pow(c12 * sdelta, 2) + math.pow(c11 * s12 - s11 * c12 * cdelta, 2))
23 x = s11 * s12 + c11 * c12 * cdelta
24 ad = math.atan2(y, x)
25 dist = ad * rad
26
27 print(round(dist))
```

### **Задача 2.2.9. Инвентаризация (10 баллов)**

В некоторой очень крупной международной корпорации решили произвести учет продукции на складах всех филиалов, распределительных центров и магазинов. Каждый склад прислал список товаров, в котором каждая строка соответствует одному товару и содержит его название и количество. Посчитайте, сколько товаров каждого вида хранится на складах, и выводите эти значения по запросам.

#### **Формат входных данных**

На вход подаются несколько списков товаров со складов ( $N < 10^3$ ). После каждого списка следует пустая строка.

В каждом списке некоторое количество записей ( $M < 10^3$ ). Одна запись записывается в отдельной строке.

Строка начинается с название товара  $s_i$  ( $1 \leq \text{len}(s_i) \leq 70$ ), которое может состоять из заглавных/строчных латинских букв, цифр и пробелов между ними. Названия товаров в одном списке не повторяются. Заканчивается строка целым числом, отделенным пробелом,  $c_i$  ( $0 \leq c_i \leq 10^6$ ) – количеством элементов.

Далее в отдельной строке записывается слово QUERIES .

После него в отдельных строках подаются  $K$  запросов  $q_i$  ( $1 \leq \text{len}(q_i) \leq 70$ ),  $K$  ( $K \leq 10^5$ ), которые могут состоять из заглавных/строчных латинских букв, цифр и пробелов между ними.

#### **Формат выходных данных**

Для каждого запроса выведите в отдельной строке единственное число – количество данного товара на всех складах. Если товар отсутствует, выведите 0.

#### **Пояснения к ответу**

Названия товаров регистрозависимы. Все названия начинаются и заканчиваются видимыми знаками (не разделителями). Нигде в исходных данных не встречаются два пробела подряд.

#### **Система оценки**



Примерно в 30% тестов названия товаров и запросы состоят из 1 слова.

Примерно в 30% тестов  $N \cdot M \geq 10^5$ .

Примерно в 40% тестов  $K \geq 10^4$

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

*Пример №1*

Стандартный ввод
blue ink 15 3 1
QUERIES
blue ink 15 3
black paper
Стандартный вывод
1
0

*Пример №2*

Стандартный ввод
robot 1 15
detail2 10
detail 1 5
professional device v4 4
detail2 12
QUERIES
robot 1
detail2
detail 2
professional device
professional device v4
MakeIt kit
robot 1 15
Стандартный вывод
0
22
0
0
4
0
0

*Пример №3*

Стандартный ввод
QUERIES
Стандартный вывод

## Решение

Основная сложность задачи – считывать поток данных до конца файла (не зная длины заранее) и разбивать строки с товарами на ключ-значение, правильно сохраняя общее количество товаров.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2 from collections import defaultdict
3
4 def parse_inventory(inventory_text):
5     inventory = defaultdict(int)
6     for line in inventory_text.splitlines():
7         words = line.split()
8         if len(words) >= 2:
9             title, quantity = ' '.join(words[:-1]), int(words[-1])
10            inventory[title] += quantity
11    return inventory
12
13 dataset = sys.stdin.read()
14
15 inventory_text, queries = [part.strip() for part in dataset.split('QUERIES')]
16 inventory = parse_inventory(inventory_text)
17
18 for item in queries.splitlines():
19     print(inventory[item])

```

### Задача 2.2.10. Теория шести рукопожатий (10 баллов)

Валентина решила проверить теорию "6 рукопожатий" в некоторой социальной сети. Согласно данной теории, все люди друзья со всеми, если не напрямую, то максимум через цепочку из 5 друзей. То есть если Маша подруга Даши, то они подруги через 1 рукопожатие. Если Даша – подруга Саши, а Саша и Маша не подруги напрямую, то они подруги через 2 рукопожатия. Если у Саши есть друг Миша, с которым не знакомы Даша и Маша, то Миша и Даша – друзья через 2 рукопожатия, а Миша и Маша – через 3 (Маша - Даша - Саша - Миша).

Валентина выгрузила список дружеских связей в формате пар уникальных никнеймов. Если пользователь  $a$  друг пользователю  $b$ , то  $b$  – друг пользователю  $a$ . Помогите Валентине проанализировать пользователей социальной сети.

#### Формат входных данных

На вход программе подается целое число  $n$  ( $2 \leq n \leq 200$ ) – количество пользователей социальной сети.

Затем в  $n$  строках подаются никнеймы. Каждый никнейм уникален, состоит из строчных латинских знаков, не короче 3 знаков и не превышает 15 знаков по длине.

Далее подается число  $m$  ( $1 \leq m \leq 10000$ ) – количество выгруженных дружеских связей.

Затем в  $m$  строках записываются пары слов – никнеймы пользователей социальной сети, которые являются прямыми (через 1 рукопожатие) друзьями. Данные пары в списке могут повторяться

### Формат выходных данных

Выведите в отдельной строке Yes, если теория "6 рукопожатий" для данной соц-сети истинна, No – в противном случае.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

### Пример №1

Стандартный ввод
8
dasha
masha
sasha
misha
dima
kolya
igor
valya
7
dasha masha
misha dima
masha sasha
kolya igor
sasha misha
dima kolya
kolya valya
Стандартный вывод
Yes

### Решение

В данной задаче можно воспользоваться алгоритмом Флойда-Уоршелл нахождения кратчайших путей между всеми парами вершин. В данном случае вершины – пользователи социальной сети, ребра – прямая связь между двумя пользователями. Вес ребра равен 1 – если есть прямая связь,  $\infty$  (в данной задаче любое число больше максимального количества пользователей) – если прямой связи нет.

Если между какими-либо двумя вершинами путь превышает 6, то выводим "No". В противном случае "Yes".

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4
5 l = dataset.split()
6 n = int(l[0])
7 m = int(l[n + 1])
8 names = l[1: n + 1]
9 pairs = l[n + 2:]
10
11 matrix = [[1000 for _ in range(n)] for _ in range(n)]
12 for i in range(n):
13     matrix[i][i] = 0
14
15 for i in range(m):
16     a = names.index(pairs[2 * i])
17     b = names.index(pairs[2 * i + 1])
18     matrix[a][b] = 1
19     matrix[b][a] = 1
20
21 for k in range(n):
22     for i in range(n):
23         for j in range(n):
24             matrix[i][j] = min(matrix[i][j], matrix[i][k] + matrix[k][j])
25
26 ans = 'Yes'
27 for i in range(n):
28     for j in range(n):
29         if matrix[i][j] > 6:
30             ans = 'No'
31             break
32
33 print(ans)

```

## 2.3. Третья попытка.

### *Задача 2.3.1. Диета (10 баллов)*

Родители научили Иру, что много есть сладкого вредно. Поэтому она решила есть не больше 8 конфет в неделю. Бабушка Ире привезла мешок с конфетами, в котором 100 конфет Красная Шапочка и 100 конфет Мишка на Севере. Ира решила выбрать 8 конфет из мешка и разложить их по дням на неделю. Ещё Ира не хочет в любой из дней оставаться без конфет. Сколькими способами она может это сделать? Порядок употребления конфет в каждый из дней не важен.

#### *Решение*

Очевидно, что девочка может выбирать любое количество конфет каждого типа, так как Красных шапочек не менее 8 и Мишек на Севере тоже.

8 конфет на 7 дней означает, что 6 дней у нее обычные (по одной конфете), а один – особенный (2 конфеты). Выбрать особенный день в неделе можно 7 способами. Количество способов выбрать конфеты в обыденные дни равно  $2^6$ , а количество

способов выбрать конфету в особенный день равно 3 (две одного вида, или две другого вида, или две разные). Так как описанные процессы выбора независимы, то количество различных комбинаций равно  $7 \cdot 2^6 \cdot 3 = 1344$ .

**Ответ:** 1344.

### ***Задача 2.3.2. Глобальные проблемы (10 баллов)***

Используя открытые официальные данные Всемирного банка (международной финансовой организации, акционерами которой являются правительства 189 стран мира, – <https://data.worldbank.org/>), отранжируйте страны по возрастанию количества выделяемого углекислого газа на душу населения (килотонн на человека) по состоянию на 2014 год.

Для получения правильного ответа скачайте и объедините две таблицы с данными (по массе выбрасываемого в атмосферу  $CO_2$  и по численности населения стран).

1. Куба
2. Танзания
3. Аргентина
4. Финляндия
5. Соединенные Штаты
6. Россия
7. Вьетнам
8. Китай
9. Саудовская Аравия
10. Канада

#### ***Решение***

Для решения задачи скачиваем две электронные таблицы: CO2 emissions (kt) – <https://data.worldbank.org/indicator/EN.ATM.CO2E.KT>, Population, total – <https://data.worldbank.org/indicator/SP.POP.TOTL>. Далее надо совместить таблицы и взять колонки за 2014 год. Для каждой страны найти отношение количества выделяемого углекислого газа к количеству людей. Далее отсортировать по возрастанию и выделить страны из списка.

**Ответ:** 2, 7, 1, 3, 8, 4, 6, 10, 5, 9.

### ***Задача 2.3.3. Произвемодульная игра (10 баллов)***

Двое юношей Петя и Витя играют в следующую игру. Петя говорит Вите три числа  $a$ ,  $b$  и  $c$ . Витя, зная, что  $c = (a \cdot x) \bmod b$  называет наименьшее возможное неотрицательное значение  $x$ , при котором  $a$ ,  $b$  и  $c$  равны названным значениям. Если  $x$  не существует, Витя говорит 0. Помогите Вите максимально быстро и безошибочно вычислять значение  $x$ .

## Формат входных данных

На вход в единственной строке через пробел подаются три целых числа  $a$ ,  $b$  и  $c$  ( $1 \leq a, b, c \leq 1000$ ).

## Формат выходных данных

Единственное число — ответ на задачу.

## Система оценки

Баллы за задачу будут начислены, если все тесты будут пройдены успешно.

### Пример №1

<b>Стандартный ввод</b>
10 7 5
<b>Стандартный вывод</b>
4

### Решение

При заданных ограничениях очевидно, что удобнее всего решать данную задачу методом перебора.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4 a, b, c = map(int, dataset.split())
5 d = 0
6 if b > c:
7     for x in range(b):
8         if (a * x) % b == c:
9             d = x
10            break
11 print(d)

```

### Задача 2.3.4. Игра Жизнь (10 баллов)

Правила этой игры просты:

- Игра Жизнь проходит на клеточном поле, которое, традиционно, называется Вселенная.
- Каждая клетка может быть живой или мёртвой (живая — \*, мёртвая — x).

- Поколения сменяются синхронно по следующим правилам:
  - в пустой (мёртвой) клетке, рядом с которой ровно три живые клетки, зарождается жизнь;
  - если у живой клетки есть две или три живые соседки, то эта клетка продолжает жить;
  - в противном случае клетка умирает от одиночества или от перенаселённости.

Для заданного начального состояния определите:

- Вечная ли Жизнь во Вселенной?
  - Если да, то укажите номер первой итерации, которая повторила любую из предыдущих.
  - Если нет, то сколько итераций осталось до смерти Вселенной
  - Под итерацией понимается процесс смены поколения во Вселенной.

### Формат входных данных

В первой строке задается целое число  $n$  ( $1 \leq n \leq 10$ ) — величина стороны Вселенной.

Далее в  $n$  строках подаются значения клеточного поля.

### Формат выходных данных

В первой строке Yes или No ответ на первый вопрос.

Во второй строке неотрицательное целое число — комментарий к ответу.

### Система оценки

Баллы за задачу будут начислены, если все тесты будут пройдены успешно.

#### Пример №1

<b>Стандартный ввод</b>
5 xxxxx xx*xx ***xx xx*xx xxxxx
<b>Стандартный вывод</b>
Yes 10

#### Пример №2

Стандартный ввод
5 *xxxx x*xxx x**xx xxxxx xxxxx
Стандартный вывод
No 5

### Решение

В данной задаче необходимо производить эмуляцию игры жизнь и сохранять игровые поля. В любом случае, в какой-то момент смена поля не будет происходить (цикл из 1 итерации) или будет наблюдаться цикличность смены игрового поля. В любом случае, эмуляция должна производиться до первого повторения поля с каким-либо предыдущим. Для реализации процесса накопления состояний поля целесообразнее использовать тип set.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 def get_next(n, s):
4     new_s = ''
5     for i in range(n):
6         for j in range(n):
7
8             idx = [(i - 1), j - 1), \
9                   ((i - 1), j), \
10                  ((i - 1), j + 1), \
11                  ((i), j - 1), \
12                  ((i), j + 1), \
13                  ((i + 1), j - 1), \
14                  ((i + 1), j), \
15                  ((i + 1), j + 1)]
16             idx = [x * n + y for (x, y) in idx if x >= 0 and x < n and
17                   y >= 0 and y < n]
18             alive = sum([1 for x in idx if s[x] == '*'])
19             if s[i * n + j] == 'x' and alive == 3:
20                 new_s += '*'
21             elif s[i * n + j] == '*' and (alive == 2 or alive == 3):
22                 new_s += '*'
23             else:
24                 new_s += 'x'
25     return new_s
26
27 dataset = sys.stdin.read()
28 d = dataset.split()
29 n = int(d[0])
30 p = ''
31 for x in d[1:]:

```



```

32     p += x
33
34     died = 'x' * n * n
35
36     s = set([died])
37
38     while True:
39         if p in s:
40             break
41         else:
42             s.add(p)
43             p = get_next(n, p)
44
45     ans = ''
46     if p == died:
47         print('No')
48     else:
49         print('Yes')
50
51     print(len(s) - 1)

```

### Задача 2.3.5. Это сдвиг? (10 баллов)

Мальчик Кирилл написал однажды на листе бумаги строчку, состоящую из больших и маленьких латинских букв, а после этого ушел играть в футбол. Когда он вернулся, то обнаружил, что его братик Дима написал под его строкой еще одну строчку такой же длины. Дима утверждает, что свою строчку он получил циклическим сдвигом строки Кирилла направо на несколько шагов (циклический сдвиг строки abcde на 2 позиции направо даст строку deabc). Однако Дима еще маленький и мог случайно ошибиться в большом количестве вычислений, поэтому Кирилл в растерянности — верить ли Диме?

#### Формат входных данных

Первые две строки содержат  $s_1$  и  $s_2$  ( $1 \leq \text{len}(s_1) = \text{len}(s_2) \leq 10^4$ ) — строки Кирилла и Димы соответственно. Строки состоят только из латинских букв.

#### Формат выходных данных

По данным строкам выведите единственное число — минимально возможный размер сдвига или  $-1$ , если Дима ошибся.

#### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

<b>Стандартный ввод</b>
ieeeeeieeeeeeeeeeeeeeeeeee ieeeeeeeeeeeeeeeeeeeeeeeee
<b>Стандартный вывод</b>
14

## Решение

Задача решается с использованием z-функции от строки имеющей следующую структуру:

$$s + \# + t + t[:-1],$$

где  $s$  – первая строка,  $t$  – вторая строка,  $t[:-1]$  – вторая строка без последнего знака,  $\#$  – не используемый в словах  $s$  и  $t$  знак.

## Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2 import string
3
4 def z_func(s):
5     i, slen = 1, len(s)
6     out = [0] * slen
7     out[0] = slen
8     while i < slen:
9         left, right = 0, i
10        while right < slen and s[left] == s[right]:
11            left += 1
12            right += 1
13        out[i] = left
14        i += 1
15    return out
16
17 dataset = sys.stdin.read()
18 s, t = dataset.split()
19 listz = z_func(s + '#' + t + t[:-1])
20
21 lt = len(t)
22 ls = len(s)
23
24 list = [idx - lt - 1 for idx in range(lt + ls + 1) if lt == listz[idx]]
25
26 if len(list) == 0:
27     list = [-1]
28
29 print(list[0])

```

## Задача 2.3.6. Пропускная способность (10 баллов)

Антенна беспроводной связи типа  $UR/2$  имеет радиус действия, равный  $R$ . Если телефон находится на расстоянии  $s \in [0, R/2]$  скорость передачи данных для одного клиента равна  $u$ . Если же  $s \in (R/2, R]$ , скорость падает до значения  $v$ . За пределами этого радиуса  $UR/2$  телефон этой антенной обслуживаться не будет.

В некотором государстве массово, почти на каждом шагу установлены антенны беспроводной связи типа  $UR/2$ . Вам нужно синхронизировать ваши данные на телефоне с облачным сервисом. С какой максимальной скоростью в заданной локации вы сможете это сделать, если технически устройством поддерживается параллельная передача данных через все доступные антенны?

### Формат входных данных

В первой строке через пробел подаются целые значения  $R, u, v$  ( $-10^8 \leq R \leq 10^8, 1 \leq v \leq u \leq 100$ ) — характеристики антенн.

В следующей строке целое число  $n$  ( $1 \leq n \leq 10^6$ ) — количество антенн.

Далее в  $n$  строках пары целых чисел  $x_i, y_i$  ( $-10^8 \leq x_i, y_i \leq 10^8$ ) — координаты антенн.

В следующей строке через пробел подаются  $X, Y$  ( $-10^8 \leq X, Y \leq 10^8$ ) — ваши координаты.

### Формат выходных данных

В единственной строке выведите целое число — скорость синхронизации в заданной локации через заданную сеть приёмников.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

Стандартный ввод
10 5 3
4
0 0
0 5
10 0
10 10
0 0
Стандартный вывод
13

#### Решение

В задаче требуется для всех антенн вычислить расстояние от передатчика до антенны и определить скорость передачи через антенну. Использовать операцию взятия квадратного корня нет необходимости, так как есть возможность вычислять все выражения в целых числах, что позволит избежать погрешностей, вызванных особенностью реализации чисел с плавающей точкой и операций над ними.

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2
3 dataset = sys.stdin.read()
4 lines = dataset.splitlines()
5 R, u, v = map(int, lines[0].split())
6 n = int(lines[1])
7
8 points = []
9
10 for l in lines[2 : 2 + n]:
11     tmp = list(map(int, l.split()))
12     points.append((tmp[0], tmp[1]))
13
14 p = list(map(int, lines[-1].split()))
15
16 s = 0
17
18 for ant in points:
19     R2_ant = sum([(ant[i] - p[i]) ** 2 for i in range(2)])
20     if 4 * R2_ant <= R * R:
21         s += u
22     elif R2_ant <= R * R:
23         s += v
24 print(int(s))

```

### Задача 2.3.7. Управление складом (10 баллов)

Анна хочет исследовать динамику использования складского помещения. У нее есть сведения о суточном изменении количества свободных ячеек, и она хочет составлять запросы и узнавать, каким образом изменяется количество свободных ячеек на временных интервалах между любыми двумя указанными датами. Помогите Анне решить эту задачу.

#### Формат входных данных

В первой строке целое число  $n$  ( $1 \leq n \leq 10^6$ ) количество дней, за которые есть наблюдения.

Во второй строке через пробел  $n$  целых чисел  $c_i$  ( $-100 \leq c_i \leq 100$ ) — значения суточных изменений.

В третьей строке целое число  $m$  ( $1 \leq m \leq 10^6$ ) — количество запросов Анны.

Далее в  $m$  строках по два числа через пробел  $b_i$  и  $e_i$  ( $0 \leq b_i \leq e_i \leq n - 1$ ) — первый и последний день отрезка, на котором необходимо произвести расчет.

#### Формат выходных данных

Для каждого из  $m$  запросов выведите два числа: количество занятых за указанное время ячеек и количество высвобожденных.

#### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

*Пояснения к ответу*

Для считывания в Python вместо `input()` используйте `sys.stdin.readline()` во избежании TL.

*Пример №1*

Стандартный ввод
1
-100
1
0 0
Стандартный вывод
0 100

*Пример №2*

Стандартный ввод
1
100
1
0 0
Стандартный вывод
100 0

*Пример №3*

Стандартный ввод
4
-10 5 -5 10
5
0 0
1 1
0 1
0 3
2 3
Стандартный вывод
0 10
5 0
5 10
15 15
10 5

*Пример №4*

Стандартный ввод
5
10 20 -30 -10 100
1
0 4
Стандартный вывод
130 40

## Решение

Заранее подсчитаем количество занятых ячеек в период от первого дня до  $i$ -го для всех дней. И отдельно аналогичным образом посчитаем количество свободных. Тогда каждый поступающий запрос можно обработать за  $O(1)$ .

### Пример программы-решения

Ниже представлено решение на языке Python3

```

1 import sys
2 from itertools import accumulate
3
4 dataset = sys.stdin.read()
5 lines = dataset.splitlines()
6
7 values = [int(v) for v in lines[1].split()]
8 queries = [tuple(int(v) for v in query.split()) for query in lines[3:]
9             if query.strip() != '']
10
11 inbound = [0] + list(accumulate(v if v > 0 else 0 for v in values))
12 outbound = [0] + list(accumulate(v if v < 0 else 0 for v in values))
13
14 answer = ''
15 for l, r in queries:
16     pos = inbound[r + 1] - inbound[l]
17     neg = outbound[r + 1] - outbound[l]
18     answer += '{} {} \n'.format(pos, -neg)
19
20 print(answer)

```

### Задача 2.3.8. Анализ текста (10 баллов)

Олеся занимается статистическим анализом текстов. Помогите ей посчитать, сколько каких слов встречается в тексте. Для этого:

1. В скачанном тексте замените все знаки, не являющиеся латинскими буквами, на пробелы.
2. Приведите текст к нижнему регистру.
3. Для каждого слова посчитайте, сколько его вхождений есть в тексте. Выведите список пар слово — количество его повторений в порядке убывания количества вхождений. Если несколько слов входят в текст равное количество раз, укажите их в лексикографическом порядке.

### Пояснения к ответу

Эталонный ответ и ваш будут сравниваться как последовательность слов без разделителей. Таким образом, ответы "answer 1" и "answer 01" будут считаться различными, а "answer 1" и "answer 1" одинаковыми. Для избежания ошибок, рекомендуем загружать ответ в виде файла в однобайтовой кодировке.

**Стандартный ввод**

Five little girls, of Five, Four, Three, Two, One:  
Rolling on the hearthrug, full of tricks and fun.

Five rosy girls, in years from Ten to Six:  
Sitting down to lessons - no more time for tricks.

Five growing girls, from Fifteen to Eleven:  
Music, Drawing, Languages, and food enough for seven!

Five winsome girls, from Twenty to Sixteen:  
Each young man that calls, I say "Now tell me which you mean!"

Five dashing girls, the youngest Twenty-one:  
But, if nobody proposes, what is there to be done?

Five showy girls - but Thirty is an age  
When girls may be engaging, but they somehow don't engage.

Five dressy girls, of Thirty-one or more:  
So gracious to the shy young men they snubbed so much before!

Five passe girls - Their age? Well, never mind!  
We jog along together, like the rest of human kind:  
But the quondam "careless bachelor" begins to think he knows  
The answer to that ancient problem "how the money goes"!

**Стандартный вывод**

five 9  
girls 9  
to 8  
the 7  
but 4  
of 4  
from 3  
one 3  
age 2  
and 2  
be 2  
for 2  
is 2  
more 2  
so 2  
that 2  
they 2  
thirty 2  
tricks 2  
twenty 2  
young 2  
along 1  
an 1  
ancient 1  
answer 1

**Стандартный вывод**

bachelor 1  
before 1  
begins 1  
calls 1  
careless 1  
dashing 1  
don 1  
done 1  
down 1  
drawing 1  
dressy 1  
each 1  
eleven 1  
engage 1  
engaging 1  
enough 1  
fifteen 1  
food 1  
four 1  
full 1  
fun 1  
goes 1  
gracious 1  
growing 1  
he 1  
hearthrug 1  
how 1  
human 1  
i 1  
if 1  
in 1  
jog 1  
kind 1  
knows 1  
languages 1  
lessons 1  
like 1  
little 1  
man 1  
may 1  
me 1  
mean 1  
men 1  
mind 1  
money 1  
much 1  
music 1  
never 1  
no 1  
nobody 1  
now 1  
on 1



**Стандартный вывод**

```
or 1
passe 1
problem 1
proposes 1
quondam 1
rest 1
rolling 1
rosy 1
say 1
seven 1
showy 1
shy 1
sitting 1
six 1
sixteen 1
snubbed 1
somehow 1
t 1
tell 1
ten 1
their 1
there 1
think 1
three 1
time 1
together 1
two 1
we 1
well 1
what 1
when 1
which 1
winsome 1
years 1
you 1
youngest 1
```

**Решение**

В данной задаче требуется обработать естественно-языковой текст. Для начала надо выделить в этом тексте последовательности из букв, то есть полностью убрать разделители, знаки препинания, цифры и др. Например, можно заменить знаки на пробел. Также в процессе обработки букв, необходимо привести их к нижнему регистру. Затем необходимо сохранить пары слово-количество повторений и отсортировать в соответствии с условием.

**Пример программы-решения**

Ниже представлена вспомогательная программа на языке Python3

```
1 import sys
2 import string
```

```

3
4 dataset = sys.stdin.read()
5 s = set([])
6
7 for x in dataset:
8     if x not in string.ascii_letters:
9         s.add(x)
10
11 t = dataset
12 for x in t:
13     for l in s:
14         t = t.replace(l, ' ')
15
16 t = t.lower()
17 t = t.split()
18
19 d = {}
20
21 for x in t:
22     if x in d.keys():
23         d[x] += 1
24     else:
25         d[x] = 1
26
27 sorted_d = sorted(d.items(), key=lambda x: x[0], reverse=False)
28 sorted_d.sort(key=lambda x: x[1], reverse=True)
29
30 tmp = ''
31
32 for x in sorted_d:
33     print(format(x[0], x[1]))

```

### Задача 2.3.9. Reverse engineering (10 баллов)

Витя написал программу, которая вычисляет функцию вида

$$f(x) = (ax^2 + bx + c) \bmod d,$$

где  $\bmod$  — остаток от деления. Для 25 различных значений аргументов из диапазона от 0 до 24 включительно он узнал, чему равна данная функция. Из этих данных он составил тесты к этой задаче, а саму функцию забыл. Автор задачи точно уверен в том, что значения  $a$ ,  $b$ ,  $c$  и  $d$  — неотрицательные целые числа, не превосходящие 10.

Помогите Вите восстановить эту функцию. Напишите программу, которая проходит все тесты.

В данной задаче нет ограничение на количество посылок. Ваша задача "смайнить" тесты и узнать, что же за функция зашита в программе. Как это сделать? Учитесь правильно ошибаться!

Для того, чтобы вы могли смотреть вердикты по всем тестам, вы обязательно должны проходить верно первый тест.

#### Формат входных данных

Целое число  $N$  ( $0 \leq N \leq 24$ ) — аргумент функции.

## Формат выходных данных

Единственное целое число — значение функции при заданном аргументе.

### Система оценки

Баллы за задачу будут начисляться пропорционально количеству успешно пройденных тестов.

#### Пример №1

<b>Стандартный ввод</b>
6
<b>Стандартный вывод</b>
4

#### Решение

Можно, конечно, попробовать решить задачу полным перебором, однако, это крайне энергозатратно, медленно и неправильно. Поэтому перебор должен быть умным. Особенность задачи в том, что участник может посмотреть на вердикты по всем тестам и может управлять вердиктами.

Например, можно печатать определенное значение функции и узнать в каких тестах оно является ответом. Так можно оценить максимальное значение функции и предположить, чему равно  $d$ .

Также можно определять значение аргумента и при определенном условии генерировать ошибку, например, Time Limit Exceeded (к примеру, while True) или Runtime Error (10/0). Так можно определить четность/нечетность аргументов, кратность любому значению или даже само значение.

Если выявить несколько пар значений аргументов и значений функций в них, можно решить систему уравнений и определить используемые коэффициенты.

### Пример программы-решения

Ниже представлено решение на языке Python3

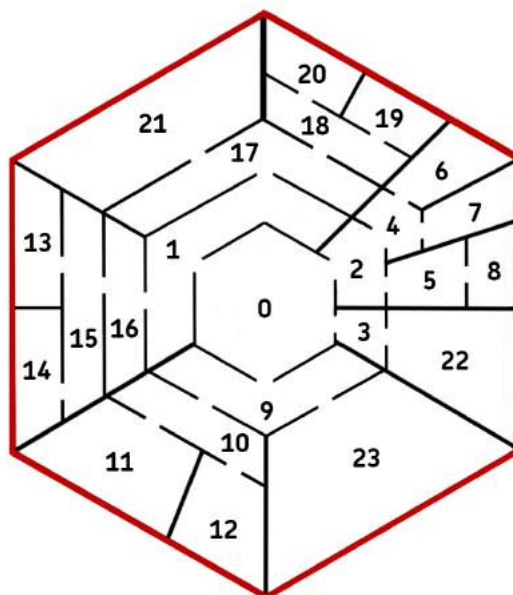
```

1 import sys
2
3 x = int(input())
4 a = 6
5 b = 3
6 c = 8
7 d = 7
8 print((a * x * x + b * x + c) % d)

```

### Задача 2.3.10. До красных стен (10 баллов)

Робот находится в центре некоторого лабиринта. Лабиринт состоит из помещений и дверей:



На схеме изображены помещения, ограниченные радиальными и шестиугольными стенами. По краям лабиринта стены покрашены в красный цвет. В шестиугольных стенах смонтированы двери между помещениями. Радиальные стены соединены с красной стеной. Дверей в красной стене нет.

Каждое помещение имеет свой уникальный номер. Номер центрального помещения, из которого стартует робот, всегда равен 0. Ни один номер не повторяется дважды. Между помещениями установлены двери. Каждая дверь связывает некоторую пару помещений. Между двумя помещениями может быть не более одной двери. Нет ни одного помещения, в которое нельзя прийти из центра и из которого нельзя выйти к красной стене, двигаясь по направлению из центра. В радиальных стенах двери отсутствуют.

Задача робота добраться до красной стены. Робот в данном лабиринте ведет себя следующим образом: каждый раз он случайно (равновероятно) выбирает дверь, ведущую в сторону от центра. В каких помещениях робот будет заканчивать работу чаще? Определите вероятности попадания робота в помещения с красной стеной.

#### Формат входных данных

Для каждого помещения с красной стеной в отдельной строке выведите вероятность попадания робота в данное помещение в следующем формате:

- номера помещений запишите в порядке возрастания;
- после каждого номера поставьте двоеточие;
- затем через пробел укажите вероятность попадания в указанное помещение. Если вероятность является целым числом, укажите это число. Если вероятность — дробное число, то запишите его в виде простой дроби  $x/y$ , где  $\text{НОД}(x, y) = 1$ .

## Система оценки

Баллы за задачу будут начислены, если все тесты будут пройдены успешно.

### Пример №1

Стандартный ввод
23
0 1
0 2
0 3
0 9
1 16
1 17
16 15
15 14
15 13
17 21
17 18
18 19
18 20
2 4
2 5
4 6
4 7
5 8
3 22
9 23
9 10
10 11
10 12
Стандартный вывод
6: 1/16
7: 1/16
8: 1/8
11: 1/16
12: 1/16
13: 1/16
14: 1/16
19: 1/32
20: 1/32
21: 1/16
22: 1/4
23: 1/8

### *Решение*

Из условия очевидно, что структура переходов между помещениями древовидная, вершина дерева помещение с номером 0, листья – помещения у красной стены. Вероятность перехода из родительской вершины в конкретную из дочерних равна  $1/n$ , где  $n$  – количество детей у родительской вершины. Собственно, вероятность

попадания в лист равно произведению вероятности попадания в его родителя, умноженное на количество детей у его родителя.

Дроби (сами вероятности) умножать при этом абсолютно неправильно, можно перемножать знаменатели и поделить в самом конце. Ограничения выставлены таким образом, что длинная арифметика не требуется.

### Пример программы-решения

Ниже представлено решение на языке Python3

```
1 from collections import defaultdict
2 import sys
3 import random
4 import re
5 import string
6
7 def dfs(graph, root, p_inv=1, visited=None, leaves=None):
8     visited = set() if visited is None else visited
9     leaves = {} if leaves is None else leaves
10
11     visited.add(root)
12     children = graph[root]
13     if len(children) == 0:
14         leaves[root] = p_inv
15     for child in children - visited:
16         dfs(graph, child, p_inv * len(children), visited, leaves)
17
18     return leaves
19
20
21 dataset = sys.stdin.read()
22 edges = [tuple(int(v) for v in edge.split()) for edge in dataset.splitlines()[1:]]
23
24 graph = defaultdict(set)
25 for src, dst in edges:
26     graph[src].add(dst)
27
28 probs = dfs(graph, 0)
29 print('\n'.join('{}: {}'.format(v, '1' if probs[v] == 1 else '1/{}'.format(probs[v]))
30         for v in sorted(probs.keys())))
```