

**Дарья Александровна Турубарова**

Краснодарский край, город Армавир

ЧОУ средняя общеобразовательная школа «Перспектива», 8 класс

**Изучение траектории бильярдного шара при помощи компьютерного моделирования**

Научный руководитель: Светлана Александровна Турубарова, учитель математики ЧОУ СОШ «Перспектива» г. Армавир

### **Аннотация**

**Целью** этого проекта является написание программы, строящей траекторию бильярдного шара в прямоугольной области, и исследование условий замкнутости и незамкнутости траекторий в зависимости от размеров бильярдного стола и угла удара.

Для выполнения исследования нужно было решить следующие **задачи**: 1) построить математическую модель нахождения траектории шара; 2) разработать алгоритм построения траектории шара; 3) написать программу, рисующую траектории; 4) изменяя начальные параметры (размеры стола и направление удара), сделать вывод об условиях замкнутости траектории; 5) сравнить полученные результаты с известными ранее.

**В результате** выполнения проекта была построена математическая модель движения шара, в частности, были наложены условия, упрощающие задачу: отсутствие трения, абсолютная упругость удара (угол падения равен углу отражения), отсутствие срединных лунок. Был разработан алгоритм построения траектории шара и реализован на языке PascalABC. В результате машинного эксперимента было установлено, что в случае целых сторон и рациональных угловых коэффициентов траектории будут замкнутыми. Если же траектории незамкнуты, то они постепенно закрашивают весь прямоугольник, что согласуется с известными результатами.

Основные **выводы** можно разбить на две группы: математические и связанные с программированием.

1. Для решения задачи нахождения траектории бильярдного шара оказалось достаточным знать уравнение прямой и уметь решать системы линейных уравнений.
2. Если угловой коэффициент линии удара пропорционален отношению сторон, то траектории будут замкнутыми, что согласуется с известным ранее результатом.
3. Если угловой коэффициент линии удара не пропорционален отношению сторон, то траектории застилают весь стол. Это означает, что рано или поздно шар попадет в срединную лунку или столкнется с другим шаром, где бы он ни лежал.
4. В силу того, что на экране конечное число пикселей, даже в случае рационального углового коэффициента наблюдался эффект незамкнутости траекторий.
5. Метод математического бильярда может быть использован для решения задач на переливания.

**Частное образовательное учреждение  
средняя общеобразовательная школа «Перспектива»  
муниципального образования город Армавир Краснодарского края**

---

**Исследовательский проект в рамках краевой научно-практической  
конференции «Эврика»**

**ИЗУЧЕНИЕ ТРАЕКТОРИИ БИЛЬЯРДНОГО ШАРА ПРИ ПОМОЩИ  
КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ**

**Исполнитель: учащаяся 8 класса Дарья Александровна Турубарова**

**Научный руководитель: Светлана Александровна Турубарова, учитель  
математики ЧОУ СОШ «Перспектива» г. Армавир**

**Краснодарский край, город Армавир**

**2020**

**Дарья Александровна Турубарова**

Краснодарский край, город Армавир

ЧОУ средняя общеобразовательная школа «Перспектива», 8 класс

### **Изучение траектории бильярдного шара при помощи компьютерного моделирования**

Научный руководитель: Светлана Александровна Турубарова, учитель математики ЧОУ

СОШ «Перспектива» г. Армавир

#### **Научная статья**

##### **Введение**

Различные математические теории своим рождением обязаны играм. В книге «Математические бильярды» Г. А. Гальперин и А. Н. Земляков пишут: «Подобно тому как азартная игра в кости вызвала к жизни «исчисление» вероятностей, игра в бильярд послужила предметом серьезных научных исследований по механике и математике» [1, с. 8]. В подтверждение этих слов в п. 2 части 1 описано применение бильярда в параллелограмме к решению задач на переливания.

В той же книге приведены теоремы о траекториях бильярдного шара и приведены иллюстрации, построенные при помощи компьютера. Целью настоящего проекта является написание программы, позволяющей строить аналогичные траектории шара в прямоугольнике, а также исследование поведения траекторий в зависимости от размеров стола, начального положения шара, и угла удара.

Перечислим задачи, решенные в рамках данного проекта с указанием их положения в тексте.

В п. 1° части 1 на основе анализа литературы сформулирована задача математического бильярда и приведен критерий замкнутости траекторий.

В п. 2° части 1 рассмотрен пример применения математического бильярда к решению задач на переливания.

В п. 1° части 2 построена математическая модель нахождения траектории шара.

В п. 2° части 2 разработан алгоритм построения траектории шара и рассмотрены основные трудности при реализации этого алгоритма на языке PascalABC.

Анализ результатов экспериментов с различными начальными данными (размеры стола и направление удара), вывод об условиях замкнутости траекторий и сравнение полученных результатов с известными ранее произведен в п. 3° части 2.

В приложении 1 приводится блок-схема алгоритма построения траектории, в Приложении 2 приведен текст программы, а в Приложении 3 – картинки траекторий, построенные при помощи этой программы.

### **Часть 1. История вопроса и постановка задачи**

1°. Бильярд – очень популярная игра во всем мире, которая требует определенной сноровки и подготовки. Чрезвычайно любил играть в бильярд Людовик XIV. Он гордился тем, как хорошо он управляется с шарами, и насколько он грациозен при этом. Опишем кратко эту игру. Бильярдный стол обычно составлен из двух квадратов, устанавливается и укрепляется по уровню и обтягивается сукном, без шва. Билия или биль – это шар, который длинной палкой, кием, надо загнать в лузу (отверстие с карманом в углах столов).

Так как учесть все факторы игры довольно сложно, в настоящем проекте мы исследуем так называемую *математическую проблему бильярда*. Она заключается в следующем: найти траекторию одного шара, считая, что трение отсутствует и соударение с бортиком абсолютно упругое (то есть угол отражения равен углу падения). Ясно, что в зависимости от начального положения шара и направления скорости, а также от размеров поля траектория шара может быть замкнутой или нет. Наша задача состоит в написании программы, строящей ломаную, по которой движется шар и исследование этой траектории в зависимости от начальных данных.

Исследование траекторий – довольно сложная задача, но в случае прямоугольной области она хорошо изучена. Так в книге [1] в гл. 3 приводятся следующие теоремы.

**Теорема 1.** *Если тангенс угла наклона  $k$  траектории соизмерим с числом  $k^* = \frac{b}{a}$  (где  $a$  и  $b$  – длина и ширина прямоугольника), то независимо от начального положения бильярдного шара его движение будет периодическим.*

**Теорема 2.** *Если число  $\frac{k}{k^*}$  – иррациональное, то любая траектория с угловым коэффициентом  $k$  всюду плотно заполняет весь прямоугольник.*

2°. Интересно, что математический бильярд можно использовать при решении задач на переливания.

**Задача.** *Имеются два сосуда — трехлитровый и пятилитровый. Нужно, пользуясь этими сосудами, получить 1, 2, 3, 4, 5, 6, 7 и 8 литров воды. В нашем распоряжении водопроводный кран и раковина, куда можно выливать воду.*



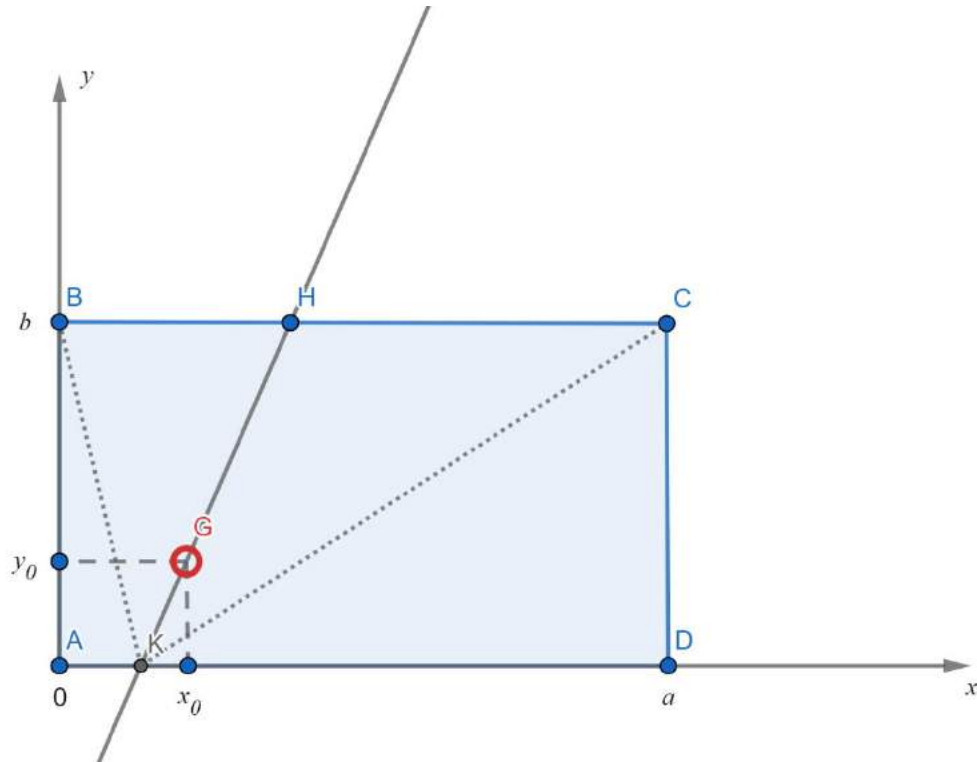


Рис. 1 (построен при помощи GeoGebra)

Очевидно, что если шар летит из точки  $G$  в точку  $H$ , то это равносильно тому, что шар летит из точки  $K$ , лежащей на стороне  $AD$ . Это означает, что начальную точку можно всегда брать на одной из сторон прямоугольника. В силу произвольности чисел  $a$  и  $b$  можно считать, что начальная точка лежит на нижней стороне.

Что нужно знать помимо начальной точки, чтобы написать уравнение прямой  $KH$ ? Нужно задать угловой коэффициент  $k_0$  (см., например, [2, с. 78]). Найдем уравнение прямой, проходящей через точку  $(x_0, y_0)$  с угловым коэффициентом  $k_0$ . Для этого в уравнение

$$y = k_0x + m$$

подставим точку  $(x_0, y_0)$ :

$$y_0 = k_0x_0 + m,$$

откуда находим  $m = y_0 - k_0x_0$ . Окончательно, уравнение прямой  $KH$  будет иметь вид

$$y = k_0x + y_0 - k_0x_0.$$

Чтобы найти координаты точки  $H$ , нужно решить одну из трех систем уравнений:

$$AB: \begin{cases} y = k_0x + y_0 - k_0x_0, \\ x = 0, \end{cases} \quad BC: \begin{cases} y = k_0x + y_0 - k_0x_0, \\ y = b, \end{cases} \quad CD: \begin{cases} y = k_0x + y_0 - k_0x_0, \\ x = a, \end{cases} \quad (1)$$

в зависимости от того, какую сторону пересекает прямая  $KH$ . Ответ на этот вопрос нам дает знание угловых коэффициентов прямых  $KB$  и  $KC$ .

Чтобы найти угловой коэффициент прямой  $KC$ , решим следующую задачу: найти уравнение прямой, проходящей через точки  $(x_K, 0)$  и  $(a, b)$ . Для этого подставим эти точки в уравнение прямой  $y = kx + m$ :

$$\begin{cases} 0 = kx_K + m, \\ b = ka + m. \end{cases}$$

Откуда находим

$$k_a = \frac{b}{a - x_K}.$$

Чтобы найти угловой коэффициент прямой  $KB$ , найдем уравнение прямой, проходящей через точки  $(x_K, 0)$  и  $(0, b)$ :

$$\begin{cases} 0 = kx_K + m, \\ b = m. \end{cases}$$

Откуда находим

$$k_b = -\frac{b}{x_K}.$$

Заметим, что случаи, когда шар летит по прямым  $KB$  или  $KC$  нас не интересуют, так как тогда шар попадет в лузу. Также нас не будет интересовать удар перпендикулярно вверх или вбок, так как тогда шар будет двигаться по одной прямой, перпендикулярной бортам.

Запишем условие, когда шар полетит на сторону  $AB$ . Очевидно, что в этом случае угловой коэффициент  $k$  будет отрицательным, но по абсолютной величине он меньше  $|k_b|$ . Это условие запишется так:  $k < 0$  и  $k > k_b$ .

Найдем теперь условие, когда шар полетит на сторону  $CD$ . Очевидно, что в этом случае угловой коэффициент  $k$  будет положительным, но меньше  $k_a$ . Это условие запишется так:  $k > 0$  и  $k < k_a$ .

Шар полетит на сторону  $BC$ , если  $k < k_b$  или  $k > k_a$ .

Решая системы (1), находим точку пересечения с

$$AB: x_H = 0, y_H = -k_0 x_K; \quad BC: x_H = \frac{b + k_0 x_K}{k_0}, y_H = b; \quad CD: x_H = a, y_H = k_0(a - x_K).$$

После нахождения координат точки  $H$ , чтобы построить отраженную траекторию, нам нужно знать новый угловой коэффициент  $k_1$ . В силу симметрии (угол отражения равен углу падения),  $k_1 = -k$ .

Построение траектории из точки  $H$  проводится аналогично.

2°. Теперь мы готовы написать алгоритм построения траектории шара на прямоугольном столе. Будем считать, что изначально шар летит с нижней стороны.

Входными параметрами являются

$a, b$  – размеры стола,

$(x_0, y_0)$  – начальное положение шара,

$k_0$  – угловой коэффициент,

$N$  – количество итераций.

Также нам понадобится переменная  $d$ , которая говорит, с какой стороны летит шар. Эта переменная принимает значение 1 на стороне  $AD$ , 2 – на стороне  $AB$ , 3 – на стороне  $BC$ , 4 – на стороне  $CD$ .

Состояние шара описывается четверкой чисел  $[x_0, y_0, k_0, d_0]$ , по этим данным функция  $\text{Point}_{d_0}$  строит новую точку  $[x_1, y_1, k_1, d_1]$ , после чего процесс повторяется сначала. Блок-схема алгоритма приведена в Приложении 1. Функции  $\text{Point1}$ ,  $\text{Point2}$ ,  $\text{Point3}$  и  $\text{Point4}$  можно посмотреть в Приложении 2.

В ходе написания программы пришлось решать проблему с тем, что функция  $\text{Line}$  принимает только целые значения, поэтому по аналогии с [3, с. 160] мы ввели свою функцию  $\text{MyLine}(x1, y1, x2, y2) = \text{Line}(\text{Round}(x1)+50, 300-\text{Round}(y1), \text{Round}(x2)+50, 300-\text{Round}(y2))$ , которая принимала произвольное число на вход и округляла до ближайшего целого.

3°. После написания программы было проведено много тестов, наиболее интересные приведены в Приложении 3. Обсудим каждый в отдельности.

1)  $a = 400, b = 200, x_0 = 30, k_0 = \sqrt{2}$ . В соответствии с теоремой 2 ч. 1 траектория должна быть незамкнутой и заполнять весь прямоугольник. Что иллюстрируют рис. 1 - 5 Приложения 3.

2)  $a = 400, b = 200, x_0 = 30, k_0 = -0,3333$ . В соответствии с теоремой 1 ч. 1 траектория должна быть замкнутой, что похоже при 20 итерациях (см. рис. 6 Приложения 3). Но при увеличении числа итераций (рис. 7 – 11 Приложения 3) создается ощущение, что они заполняют всю плоскость. Чтобы объяснить это противоречие, было решено учесть два фактора: во-первых, если траектория замкнутая, но расстояние между линиями очень маленькое, то глаз может этого не заметить, во-вторых, при компьютерных вычислениях может появляться погрешность и координаты одной точки могут быть очень близкими к координатам исходной, но все-таки не совпадать с ней. Для проверки последней гипотезы в программу было добавлено сравнение координат с точностью  $10^{-9}$ . Оказалось, что период



траектории равен 16 666, но уже при 10 000 итераций фигура кажется закрашенной. То есть траектория является замкнутой и не заполняет весь прямоугольник, но зазоры не видно невооруженным глазом.

Заметим, что в предыдущем случае ( $k_0 = \sqrt{2}$ ) даже при 20 000 итераций не находится точки близкой к начальной с погрешностью  $10^{-9}$ .

3)  $a = 400, b = 200, x_0 = 30, k_0 = 2$ . На рис. 12 Приложения 3 даже при 10 000 итерациях отчетливо видно замкнутую траекторию. То же самое при  $k_0 = 0,48$  (рис. 13 Приложения 3).

4) Далее мы изменили размеры области:  $a = 500, b = 200, x_0 = 30, k_0 = 0,48$ . Рис. 14 Приложения 3 показывает замкнутую траекторию, так как  $\frac{k}{k^*} = \frac{0,48}{2,5} = 0,192$  – рациональное число.

5) В заключение мы взяли иррациональной длину прямоугольника:  $a = 500, b = \sqrt{2} * 100, x_0 = 30, k_0 = 0,48$ . Результаты соответствуют случаю иррационального углового коэффициента (см. рис. 15 -17 Приложения 3).

### **Заключение**

В результате выполнения проекта была построена математическая модель движения шара, в частности, были наложены условия, упрощающие задачу: отсутствие трения, абсолютная упругость удара (угол падения равен углу отражения), отсутствие срединных лунок. Был разработан алгоритм построения траектории шара и реализован на языке PascalABC. В результате машинного эксперимента было установлено, что в случае целых сторон и рациональных угловых коэффициентов траектории будут замкнутыми. Если же траектории незамкнуты, то они постепенно закрашивают весь прямоугольник, что согласуется с известными результатами.

Основные выводы можно разбить на две группы: математические и связанные с программированием.

1. Для решения задачи нахождения траектории бильярдного шара оказалось достаточным знать уравнение прямой и уметь решать системы линейных уравнений.
2. Если угловой коэффициент линии удара пропорционален отношению сторон, то траектории будут замкнутыми, что согласуется с известным ранее результатом.
3. Если угловой коэффициент линии удара не пропорционален отношению сторон, то траектории застилают весь стол. Это означает, что рано или поздно шар попадет в срединную лунку или столкнется с другим шаром, где бы он ни лежал.

4. В силу того, что на экране конечное число пикселей, даже в случае рационального углового коэффициента наблюдался эффект незамкнутости траекторий.
5. Метод математического бильярда может быть использован для решения задач на переливания.

### **Литература**

1. Г. А. Гальперин, А. Н. Земляков. Математические бильярды (бильярдные задачи и смежные вопросы математики и механики). – М.: Наука, 1990.
2. Ю. Н. Макарычев, Н. Г. Миндюк, К. И. Нешков, С. Б. Суворова. Алгебра. 7 класс: учебник для общеобразовательных учреждений / Под ред. С. А. Теляковского. – М.: Просвещение, 2013.
3. В. Рубанцев. Развивающее программирование. Занимательные проекты на Паскале. / <http://pascalabc.net/downloads/Books/Rubantsev/InterestProjProjects.pdf>

Дарья Александровна Турубарова

Краснодарский край, город Армавир

ЧОУ средняя общеобразовательная школа «Перспектива», 8 класс

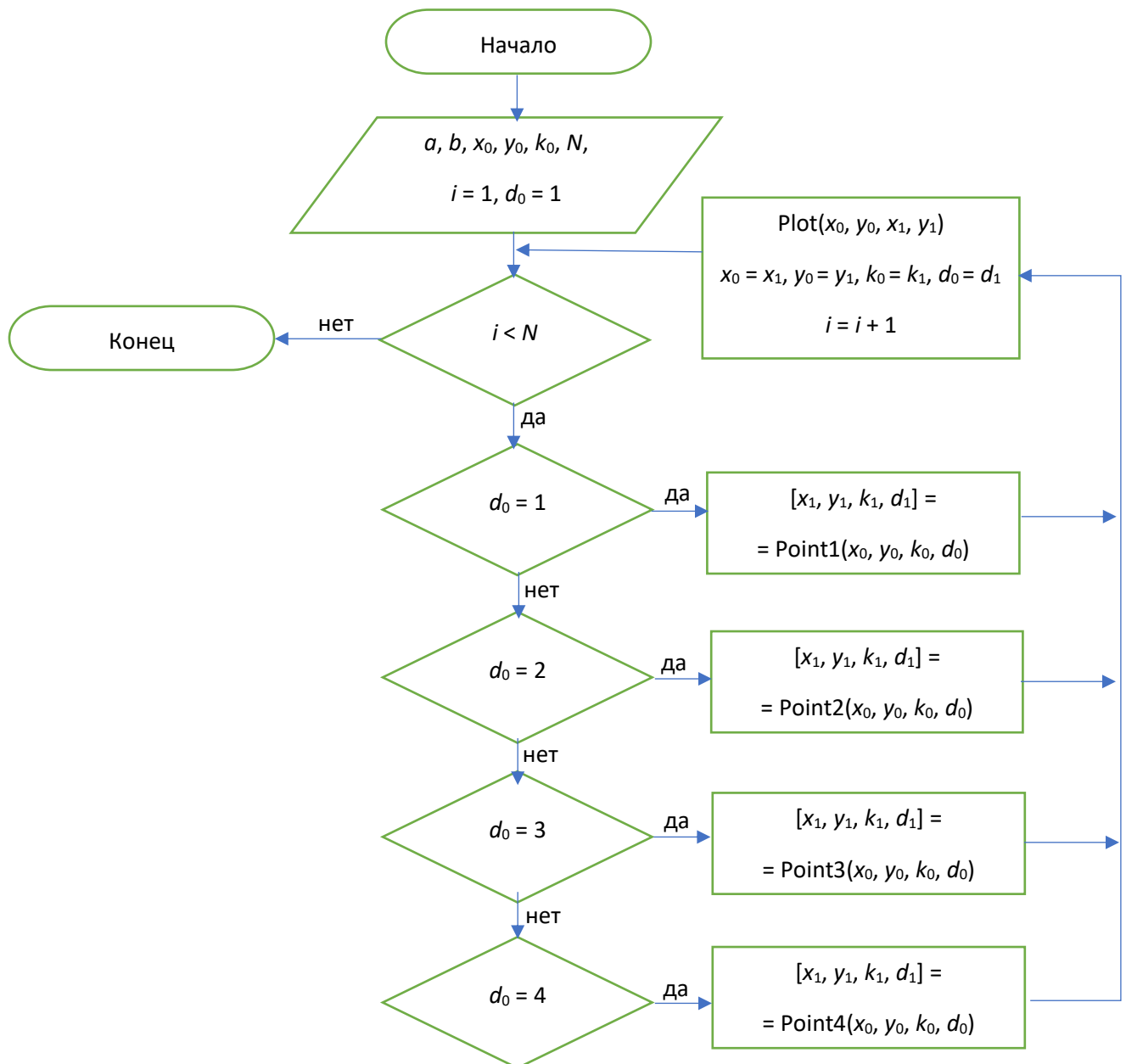
**Изучение траектории бильярдного шара при помощи компьютерного моделирования**

Научный руководитель: Светлана Александровна Турубарова, учитель математики ЧОУ

СОШ «Перспектива» г. Армавир

## Приложения

### Приложение 1. Блок-схема



## Приложение 2. Программа

```

program biliard;
uses
  GraphABC;
type mytype = array[1..4] of real;
const a=400.0;
      b=200.0;
var i, C : Integer;
      M, N : mytype;
procedure MyLine(x1,y1,x2,y2:real);
begin
  Line(Round(x1)+50,300-Round(y1),Round(x2)+50,300-Round(y2));
end;
function Point1(T:mytype):mytype;
var F:mytype;
      k1, k2:real;
begin
  k1:=(T[2]-b)/(T[1]-a);
  k2:=(T[2]-b)/(T[1]);
  If (T[3]<k1) AND (T[3]>0) Then
  begin
    F[1]:=a;
    F[2]:=T[3]*a+T[2]-T[3]*T[1];
    F[3]:=-T[3];
    F[4]:=4;
  end;
  If (T[3]>k1) OR (T[3]<k2) Then
  begin
    F[1]:=(b-T[2]+T[3]*T[1])/T[3];
    F[2]:=b;
    F[3]:=-T[3];
  end;
end;

```

```
F[4]:=3;
end;
If (T[3]>k2) AND (T[3]<0) Then
begin
  F[1]:=0;
  F[2]:=T[2]-T[3]*T[1];
  F[3]:=-T[3];
  F[4]:=2;
end;
Point1:=F;
end;

function Point2(T:mytype):mytype;
var F:mytype;
    k1, k2:real;
begin
  k1:=(b-T[2])/a;
  k2:=-T[2]/a;
  If (T[3]<0) AND (T[3]<k2) Then
  begin
    F[1]:=-T[2]/T[3];
    F[2]:=0;
    F[3]:=-T[3];
    F[4]:=1;
  end;
  If (T[3]>k1) AND (T[3]>0) Then
  begin
    F[1]:=(b-T[2])/T[3];
    F[2]:=b;
    F[3]:=-T[3];
    F[4]:=3;
  end;
  If (T[3]>k2) AND (T[3]<k1) Then
```

```

begin
  F[1]:=a;
  F[2]:=T[3]*a+T[2];
  F[3]:=-T[3];
  F[4]:=4;
end;
Point2:=F;
end;
function Point3(T:mytype):mytype;
var F:mytype;
    k1, k2:real;
begin
  k1:=b/T[1];
  k2:=b/(T[1]-a);
  If (T[3]<k1) AND (T[3]>0) Then
  begin
    F[1]:=0;
    F[2]:=-T[3]*T[1]+b;
    F[3]:=-T[3];
    F[4]:=2;
  end;
  If (T[3]>k1) OR (T[3]<k2) Then
  begin
    F[1]:=(T[3]*T[1]-b)/T[3];
    F[2]:=0;
    F[3]:=-T[3];
    F[4]:=1;
  end;
  If (T[3]>k2) AND (T[3]<0) Then
  begin
    F[1]:=a;
    F[2]:=T[3]*(a-T[1])+b;
    F[3]:=-T[3];

```

```
F[4]:=4;
end;
Point3:=F;
end;

function Point4(T:mytype):mytype;
var F:mytype;
    k1, k2:real;
begin
    k1:=T[2]/a;
    k2:=(T[2]-b)/a;
    If (T[3]>0) AND (T[3]>k1) Then
    begin
        F[1]:=(T[3]*a-T[2])/T[3];
        F[2]:=0;
        F[3]:=-T[3];
        F[4]:=1;
    end;
    If (T[3]<k1) AND (T[3]>k2) Then
    begin
        F[1]:=0;
        F[2]:=-T[3]*a+T[2];
        F[3]:=-T[3];
        F[4]:=2;
    end;
    If (T[3]<k2) AND (T[3]<0) Then
    begin
        F[1]:=(T[3]*a+b-T[2])/T[3];
        F[2]:=b;
        F[3]:=-T[3];
        F[4]:=3;
    end;
    Point4:=F;
end;
```

```
end;

begin
MyLine(0,0,a,0);
MyLine(0,0,0,b);
MyLine(0,b,a,b);
MyLine(a,0,a,b);
//-----closed-----
M[1]:=20;
M[2]:=10;
M[3]:=2;
M[4]:=1;
//-----full-----
//M[1]:=20;
//M[2]:=10;
//M[3]:=-0.3333;
//M[4]:=1;
N:=Point1(M);
writeln(N[1],N[2]);
MyLine(M[1],M[2],N[1],N[2]);
C:=12000
for i:=1 to C do
begin
if N[4]=1 then M:=Point1(N);
if N[4]=2 then M:=Point2(N);
if N[4]=3 then M:=Point3(N);
if N[4]=4 then M:=Point4(N);
MyLine(M[1],M[2],N[1],N[2]);
N:=M;
end;
end.
```



### Приложение 3. Результаты компьютерного моделирования

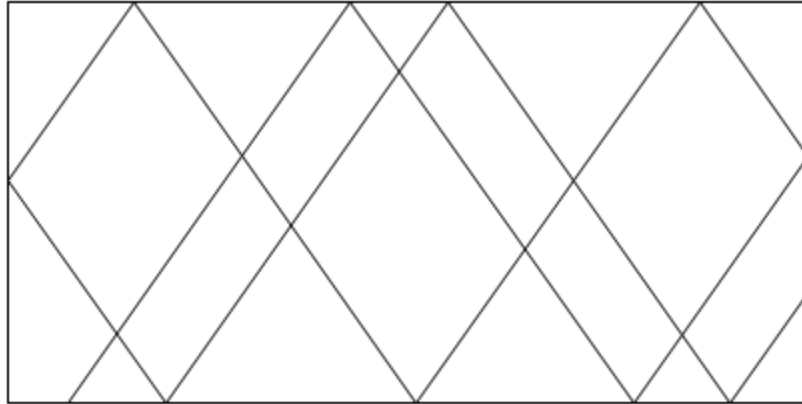


Рис. 1 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=\sqrt{2}$ ,  $N=10$ )

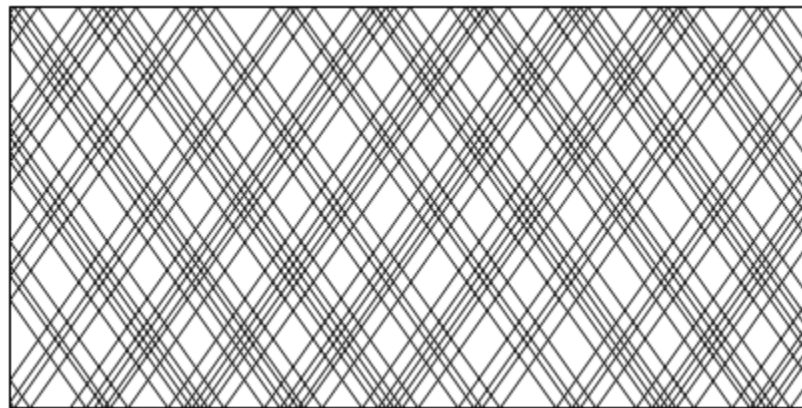


Рис. 2 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=\sqrt{2}$ ,  $N=120$ )

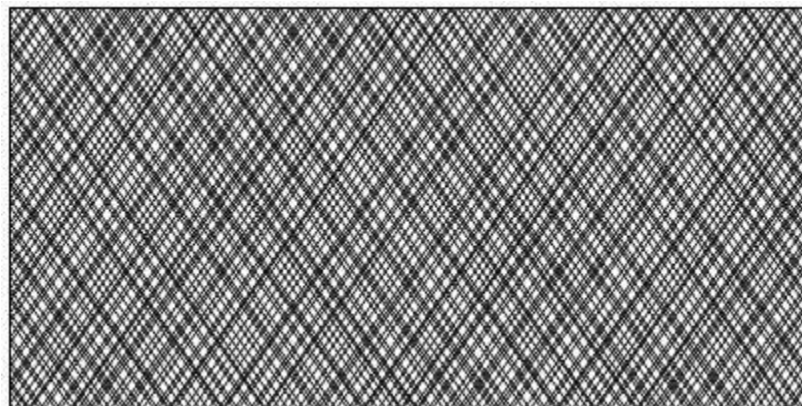
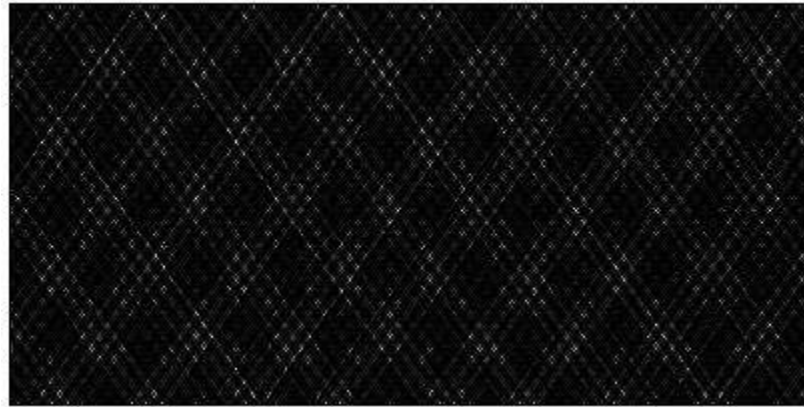


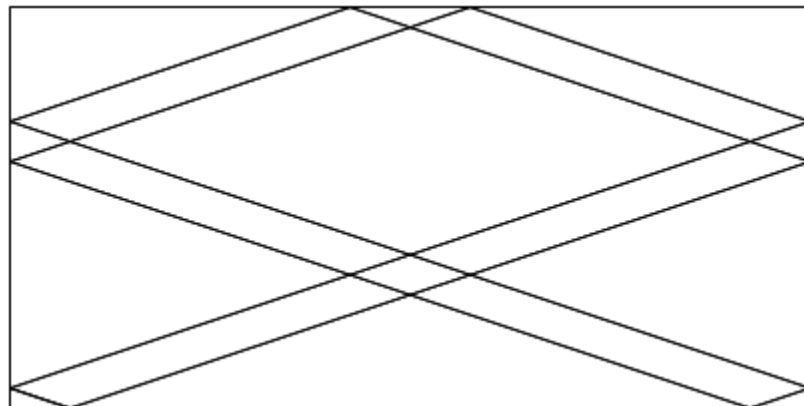
Рис. 3 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=\sqrt{2}$ ,  $N=320$ )



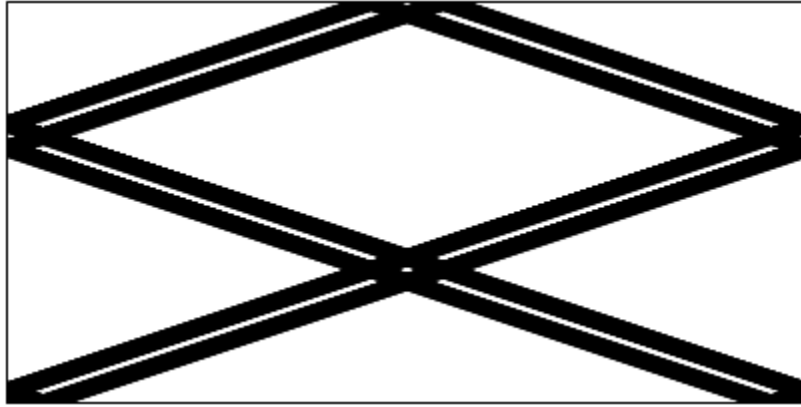
*Рис. 4* ( $a=400, b=200, x_0=30, y_0=0, k_0=\sqrt{2}, N=1200$ )



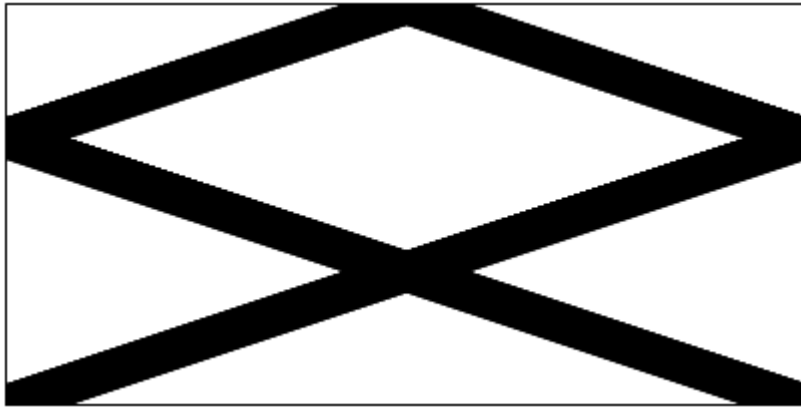
*Рис. 5* ( $a=400, b=200, x_0=30, y_0=0, k_0=\sqrt{2}, N=2000$ )



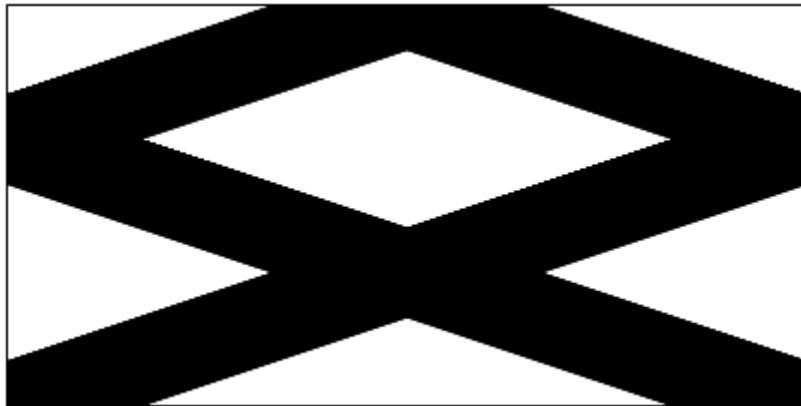
*Рис. 6* ( $a=400, b=200, x_0=30, y_0=0, k_0=-0.3333, N=20$ )



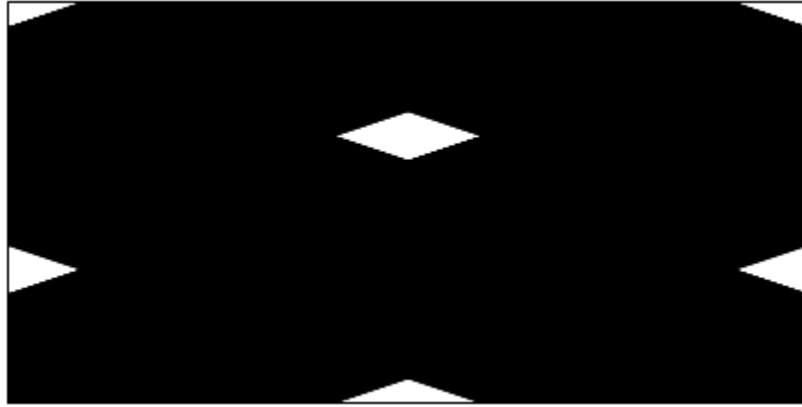
*Рис. 7 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=-0.3333$ ,  $N=1000$ )*



*Рис. 8 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=-0.3333$ ,  $N=2000$ )*



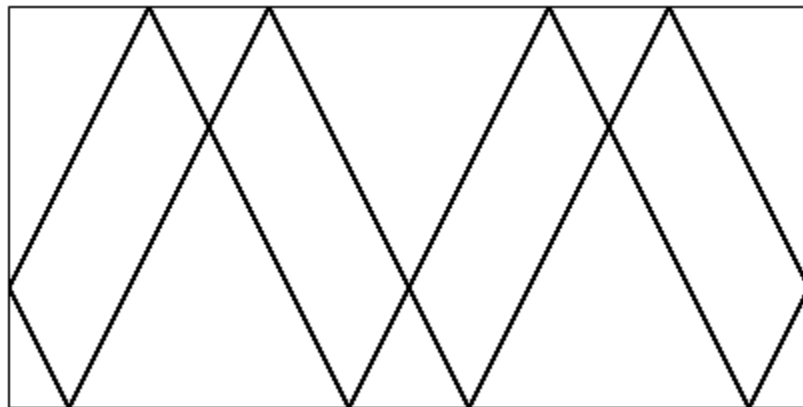
*Рис. 9 ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=-0.3333$ ,  $N=4000$ )*



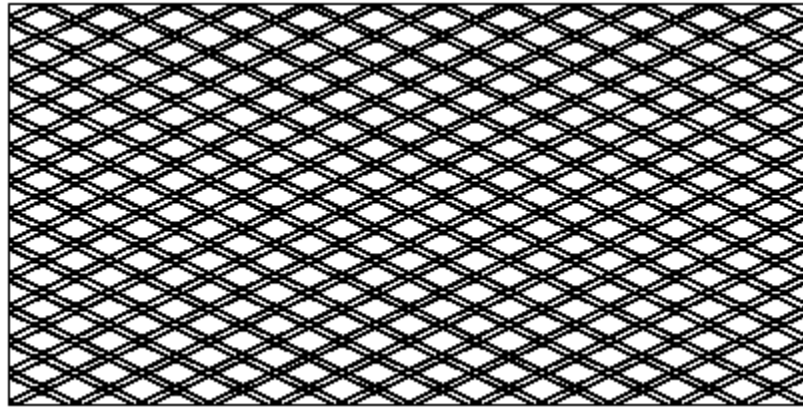
*Puc. 10* ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=-0.3333$ ,  $N=8000$ )



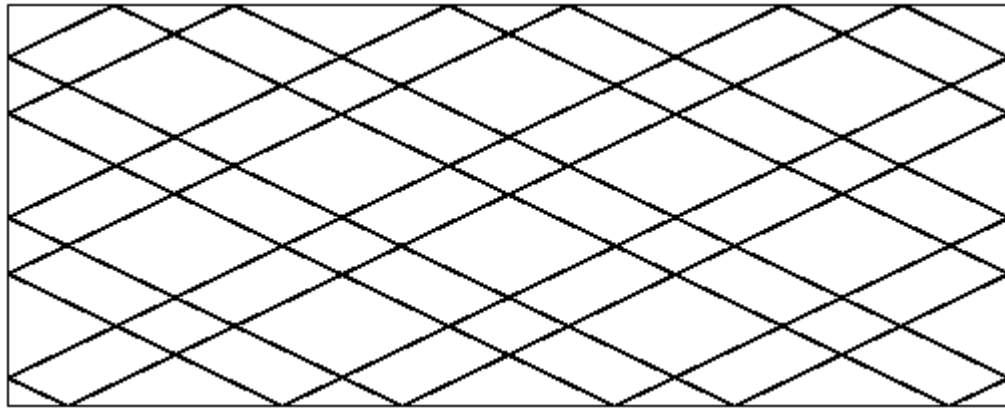
*Puc. 11* ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=-0.3333$ ,  $N=10000$ )



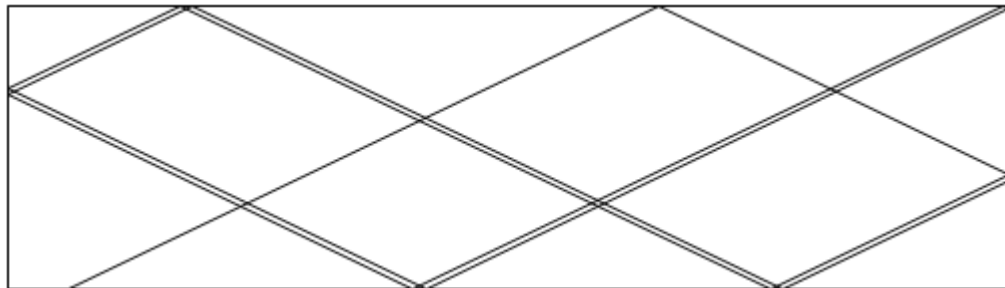
*Puc. 12* ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=2$ ,  $N=10000$ )



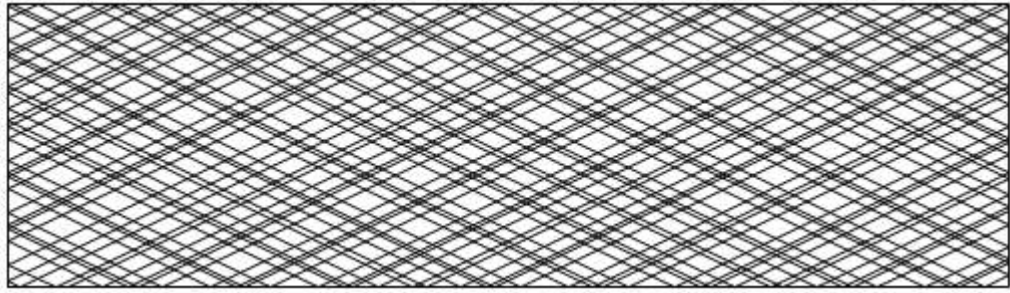
*Puc. 13* ( $a=400$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=0.48$ ,  $N=10000$ )



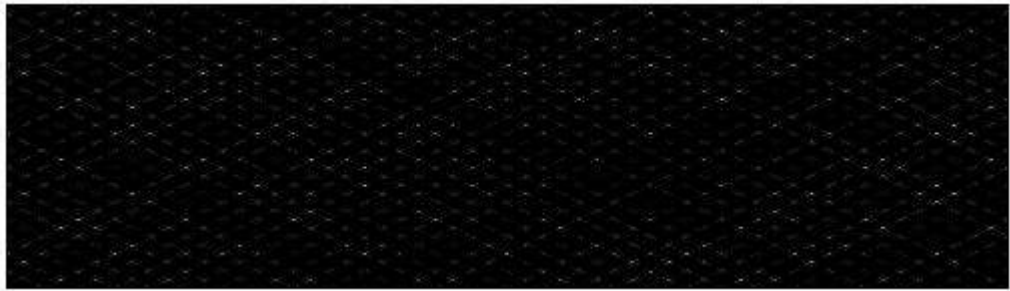
*Puc. 14* ( $a=500$ ,  $b=200$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=0.48$ ,  $N=10000$ )



*Puc. 15* ( $a=500$ ,  $b=\sqrt{2}*100$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=0.48$ ,  $N=12$ )



*Puc. 16* ( $a=500$ ,  $b=\sqrt{2}*100$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=0.48$ ,  $N=120$ )



*Puc. 17* ( $a=500$ ,  $b=\sqrt{2}*100$ ,  $x_0=30$ ,  $y_0=0$ ,  $k_0=0.48$ ,  $N=1200$ )