

Власенко Илона Сергеевна

аспирантка, Донецкий государственный университет, г. Донецк

МАТЕМАТИКА ЕГЭ, КОТОРУЮ МОЖНО УВИДЕТЬ:

ВИЗУАЛИЗАЦИЯ ЗАДАЧ С ПАРАМЕТРОМ С ПОМОЩЬЮ PYTHON

Аннотация. В статье рассматривается способ объяснения задач с параметром через визуализацию движения графиков. Такие задачи часто воспринимаются учащимися как наиболее сложные, потому что в них нужно работать не только с переменной, но и с дополнительной величиной, от которой меняется количество решений. Визуальный подход позволяет заменить абстрактное рассуждение на наглядную модель: параметр становится ползунком, график начинает двигаться, а моменты касания, пересечения и исчезновения корней становятся видимыми. На примере уравнения экзаменационного типа показано, как связать аналитическое решение через дискриминант с графической интерпретацией и как подготовить простую визуализацию средствами Python. Отдельное внимание уделяется тому, как такие визуализации можно использовать в учебной статье, интерактивной презентации или онлайн-материале для подготовки к экзамену.

Ключевые слова: профильная математика, графический метод, визуализация, Python, Matplotlib, Plotly, интерактивная модель, квадратное уравнение.

Annotation. The article examines a method for explaining problems involving parameters through the visualization of graph movement. Such problems are often perceived by students as the most difficult, because they require working not only with a variable but also with an additional quantity, the value of which affects the number of solutions. The visual approach allows abstract reasoning to be replaced with a visual

model: the parameter becomes a slider, the graph starts moving, and the moments of tangency, intersection, and disappearance of roots become visible. Using an exam-type equation as an example, the article shows how to link the analytical solution via the discriminant with the graphical interpretation and how to prepare a simple visualization using Python. Special attention is paid to how such visualizations can be used in a research article, an interactive presentation, or online study materials for exam preparation.

Keywords: advanced mathematics, graphical method, visualization, Python, Matplotlib, Plotly, interactive model, quadratic equation.

Введение

Задачи с параметром занимают особое место в школьной математике. В обычном уравнении нужно найти неизвестное число, а в задаче с параметром требуется понять, как ответ меняется при разных значениях дополнительной величины. Поэтому ученик сталкивается не с одним уравнением, а фактически с целым семейством уравнений.

Именно это делает тему сложной: решение перестает быть линейным набором вычислений. Нужно одновременно держать в голове несколько случаев, следить за границами, не потерять особые значения параметра и правильно объяснить, почему именно при них меняется количество решений.

Но у этой сложности есть другая сторона. Задачи с параметром очень хорошо визуализируются. Если представить параметр как ручку настройки, то формула превращается в движущийся график. В таком подходе корни уравнения становятся точками пересечения с осью, особые значения параметра - моментами касания, а отсутствие решений – ситуацией, когда график не пересекает нужную линию.

Основная часть

1. Почему задачи с параметром кажутся сложными

Главная трудность состоит в том, что параметр меняет саму структуру задачи. Например, квадратное уравнение может иметь два корня, один корень или не иметь действительных корней. При этом все три ситуации могут возникать в одной и той же формуле, если менять значение параметра.

В традиционном решении ученик обычно строит цепочку условий: вычисляет дискриминант, сравнивает его с нулём, решает неравенства относительно параметра. Это правильный и необходимый путь, но на первых этапах он может выглядеть как набор формальных операций. Визуализация помогает увидеть смысл этих операций.

2. Параметр как движение графика

Рассмотрим уравнение экзаменационного типа:

$$x^2 - 2ax + 3 = 0$$

При фиксированном значении a это обычное квадратное уравнение. Но если a изменяется, меняется и положение параболы $y = x^2 - 2ax + 3$. Вопрос о количестве корней превращается в вопрос о том, сколько раз парабола пересекает ось Ox .

Аналитически всё сводится к дискриминанту: $D = (-2a)^2 - 4 \cdot 1 \cdot 3 = 4a^2 - 12 = 4(a^2 - 3)$. Значит, два корня существуют при $|a| > \sqrt{3}$, один корень - при $|a| = \sqrt{3}$, а действительных корней нет при $|a| < \sqrt{3}$.

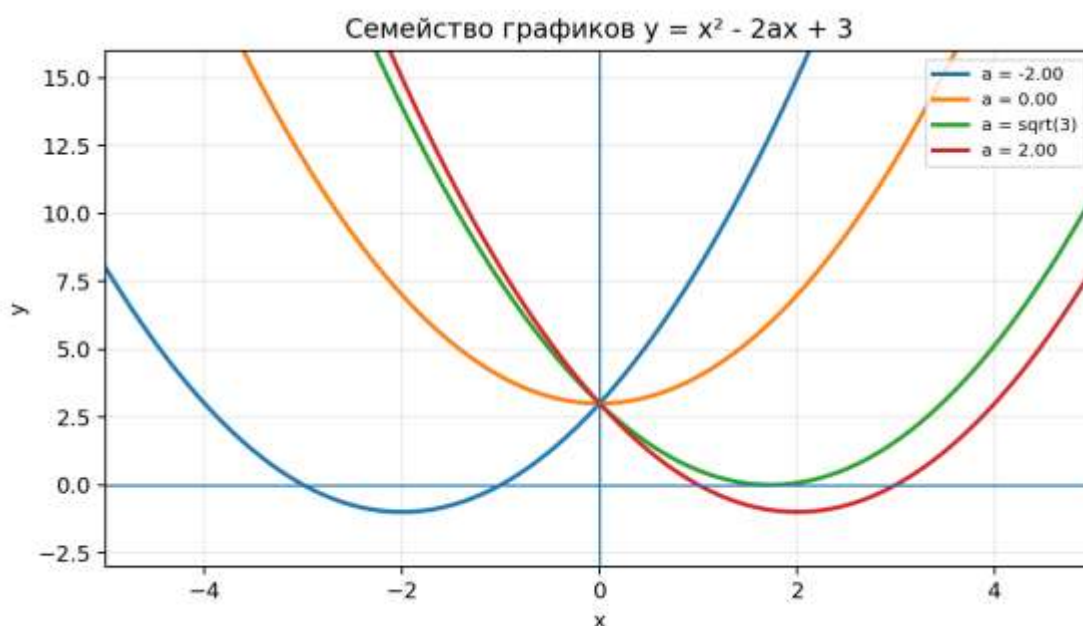


Рис. 1. При изменении параметра a меняется положение параболы и количество её пересечений с осью Ox .

3. Как связать рисунок и решение

Визуализация не заменяет доказательство, но помогает построить его. На графике видно, что при некоторых значениях параметра парабола располагается выше оси Ox и не имеет пересечений. Затем наступает предельный момент: парабола касается оси. После этого появляются две точки пересечения.

Этот предельный момент и есть случай $D = 0$. Таким образом, дискриминант перестаёт быть механической формулой и получает геометрический смысл: он показывает, есть ли у параболы пересечения с осью Ox и сколько их.

Условие	Геометрический смысл	Количество корней
$D > 0$	парабола пересекает ось Ox в двух точках	2

$D = 0$	парабола касается оси Ox	1
$D < 0$	парабола не пересекает ось Ox	0

4. Код на Python для статической визуализации

Данная программа представляет собой интерактивное веб-приложение, предназначенное для визуализации квадратной функции с параметром и исследования количества её действительных корней. Приложение написано на языках HTML, CSS и JavaScript и использует элемент Canvas для построения графика.

В основе программы лежит функция $f(x) = x^2 - 2ax + 3$, где a – изменяемый параметр. Пользователь может изменять значение параметра с помощью ползунка, после чего график автоматически перестраивается.

HTML-код создаёт интерфейс страницы: заголовок, область для рисования графика, ползунок для изменения параметра a , блок вывода информации о корнях и краткую подсказку. Основным элементом является `<canvas id="plot">`, на котором отображается координатная плоскость и парабола.

Объект `world` задаёт границы координатной системы. Функции `xToPx()` и `yToPx()` преобразуют математические координаты в координаты пикселей Canvas.

Функция `f(x, a)` вычисляет значение квадратной функции. Построение сетки и координатных осей выполняет функция `drawGrid()`, а функция `drawCurve(a)` рисует параболу, вычисляя множество точек на заданном промежутке.

Функция `draw()` объединяет все этапы работы программы:

- изменение размеров Canvas;
- получение текущего значения параметра;

- вычисление дискриминанта;
- построение сетки;
- построение графика;
- отображение точек пересечения с осью Ox ;
- вывод текстовой информации о количестве корней.

Событие `input` у ползунка обеспечивает обновление графика в реальном времени, поэтому пользователь сразу видит, как изменение параметра влияет на положение параболы и количество решений уравнения.

Визуализация особенно полезна в тех случаях, когда ученик знает формулу, но не понимает, почему она работает. В задачах с параметром рисунок позволяет увидеть переходы между случаями: было два решения, стало одно, затем решений не стало. Такой переход трудно почувствовать только через алгебраические неравенства.

Кроме того, программирование добавляет к математике исследовательский характер. Ученик может не просто получить готовый график, а менять параметр сам, проверять гипотезы, замечать закономерности и затем переводить наблюдения на язык строгого решения.

Выводы

Задачи с параметром становятся заметно понятнее, если рассматривать их как движение графиков. В этом случае параметр перестаёт быть абстрактным символом и превращается в управляемую величину, от которой зависит положение кривой, количество точек пересечения и вид ответа.

Python даёт удобные инструменты для такой работы: с помощью `Matplotlib` можно быстро построить статические иллюстрации, с помощью анимации – показать изменение графика во времени, а с помощью интерактивных библиотек или веб-технологий – создать модель с ползунками.

Главный вывод состоит в том, что визуализация не отменяет строгого решения, а делает его более осмысленным. Она помогает увидеть, почему появляются условия, где возникают особые значения параметра и как связаны алгебраические вычисления с геометрической картиной. Поэтому тема визуализации задач с параметром хорошо подходит для публикации: она соединяет школьную математику, программирование и современный способ объяснения сложных идей.

Литература

1. Бродский Я. С., Павлов А. Л. Квадратные уравнения и их применения. Пособие для дополнительного изучения математики обучающимися 8-9 классов. — 59 с.
2. Мэттиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017. — 496 с
3. Визуализация «Количество корней уравнений» [Электронный ресурс]. - Режим доступа: URL: <https://codepen.io/Ilona-Vlasenko/pen/KwaZGZg>
4. Manim Documentation [Электронный ресурс]. - Режим доступа: URL: <https://docs.manim.community/en/stable/> (04.08.2026).
5. Pyplot Documentation [Электронный ресурс]. - Режим доступа: URL: https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.html (04.08.2026).

Literature

1. Brodsky Ya. S., Pavlov A. L. Quadratic Equations and Their Applications. A guide for additional study of mathematics for students in grades 8–9. — 59 p.
2. Matthes E. Learning Python. Game Programming, Data Visualization, Web Applications. — St. Petersburg: Piter, 2017. — 496 p.
3. Visualization “Number of Roots of Equations” [Electronic resource]. - Access mode: URL: <https://codepen.io/Ilona-Vlasenko/pen/KwaZGZg>

4. Manim Documentation [Electronic resource]. - Access mode: URL:
<https://docs.manim.community/en/stable/> (04.08.2026).

5. Pyplot Documentation [Electronic resource]. - Access mode: URL:
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.html (04.08.2026).