

Доказательства свойств для множества Мандельброта

Формула: $z_{n+1} = z_n^2 + c, \quad z_0 = 0$

1 СВОЙСТВО

Симметрия относительно вещественной оси

Доказательство:

Пусть дана последовательность $\{z_n\}$ для параметра c

$$z_0 = 0, z_{n+1} = z_n^2 + c$$

По определению, $c \in M$ тогда и только тогда, когда последовательность $\{|z_n|\}$ ограничена.

Рассмотрим последовательность $\{k_n\}$ для сопряженного параметра \bar{c} :

$$k_0 = 0$$

$$k_{n+1} = k_n^2 + \bar{c}$$

Нужно доказать, что $k_n = \overline{z_n}$ для всех $n \geq 0$. Используем метод математической индукции

1. База индукции:

$$k_0 = 0 \quad \overline{z_0} = \overline{0} = 0$$

Мы видим, что $k_0 = \overline{z_0}$. База индукции есть

2. Индукционный переход:

Допустим, что $k_i = \overline{z_i}$. Рассмотрим $k_{i+1} = k_i^2 + \bar{c}$:

Используя индуктивное предположение подставим $\overline{z_i} = k_i$:

$$k_{i+1} = (\overline{z_i})^2 + \bar{c}$$

Так как $\overline{c_1 + c_2} = \overline{c_1} + \overline{c_2}$, а $\overline{z^2} = \overline{z}^2$ Получаем, что

$$k_{i+1} = \overline{z_i^2 + c}$$

Таким образом

$$k_i = \overline{z_i}$$

Мы доказали, что последовательность для \bar{c} является последовательностью сопряженных значений для c : $k_n = \bar{z}_n$.

Сравним их модули:

$$|k_n| = |\bar{z}_n|$$

Так как $|\bar{z}_n| = |z_n|$ получаем, что $|k_n| = |z_n|$

Таким образом, последовательность $\{|z_n|\}$ ограничена тогда и только тогда, когда последовательность $\{|k_n|\}$ ограничена.

По определению множества Мандельброта:

$$c \in M \iff \{|z_n|\} \text{ ограничена} \iff \{|k_n|\} \text{ ограничена} \iff \bar{c} \in M$$

Это доказывает симметрию M относительно вещественной оси.

2 свойство

Если $|c| > 2$, то c не принадлежит множеству Мандельброта ($c \notin M$).

Определение: $c \in M$ тогда и только тогда, когда последовательность $z_{n+1} = z_n^2 + c$, где $z_0 = 0$, остается ограниченной, то есть $\{|z_n|\}$ ограничена.

При $|c| > 2$. Докажем, что $\{|z_n|\}$ неограничена.

Докажем, что если $|z_n|$ достигает $|c|$, то последовательность начинает строго монотонно расти.

Рассмотрим

$$|z_{k+1}| = |z_k^2 + c| \geq |z_k^2 - c| \geq |z_k^2| - |c|$$

Нам необходимо избавиться от внешнего модуля, доказав, что внутреннее выражение неотрицательно:

$$|z_k^2| - |c| \geq 0$$

Мы ведем доказательство при условии, что $|z_k| \geq |c|$.

Поскольку $|c| > 2$, то $|z_k| \geq |c| \geq 2$. Умножим неравенство $|z_k| \geq |c|$ на $|z_k|$:

$$|z_k|^2 \geq |c| \cdot |z_k|$$

Поскольку $|z_k| \geq 2 > 1$, то $|z_k| \cdot |c| > |c|$. Следовательно, $|z_k|^2 > |c|$, что означает

$$|z_k|^2 - |c| \geq 0$$

Таким образом, внешний модуль снимается строго:

$$|z_{k+1}| \geq |z_k|^2 - |c|$$

Доказательство монотонного роста

Нам нужно доказать, что если $|z_k| \geq |c|$, то $|z_{k+1}| > |z_k|$. Это эквивалентно неравенству:

$$|z_k|^2 - |c| > |z_k| \iff |z_k| (|z_k| - 1) > |c|$$

Поскольку $|z_k| \geq |c|$ и $|c| > 2$:

1. Оценка первого множителя: $|z_k| \geq |c|$.
2. Оценка второго множителя: $|z_k| - 1 \geq |c| - 1$. Поскольку $|c| > 2$, то $|c| - 1 > 1$.

Подставим минимальные оценки:

$$|z_k| (|z_k| - 1) > |c| \cdot 1 = |c|$$

Вывод:

Если $|z_k| \geq |c|$, то $|z_{k+1}| > |z_k|$ и $|z_{k+1}| > |c|$.

Доказательство неограниченности

Поскольку $|c| > 2$, определим константу λ :

$$\lambda = |c| - 1$$

Очевидно, $\lambda > 1$.

Используя доказанное выше неравенство $|z_{n+1}| \geq |z_n|^2 - |c|$, мы выделяем множитель $|z_n|$:

$$|z_{n+1}| \geq |z_n| (|z_n| - 1) + (|z_n| - |c|)$$

Поскольку $|z_n| \geq |c|$ для $n \geq 1$:

1. Отбрасываем положительный член $|z_n| - |c| \geq 0$:

$$|z_{n+1}| \geq |z_n| (|z_n| - 1)$$

2. Оцениваем множитель:

Так как $|z_n| \geq |c|$, то $|z_n| - 1 \geq |c| - 1 = \lambda$.

$$|z_{n+1}| \geq |z_n| \cdot \lambda$$

Мы начинаем с первого члена $|z_1| = |c|$ (который удовлетворяет условию $|z_1| \geq |c|$), и применяем неравенство $|z_{k+1}| \geq \lambda |z_k|$ последовательно

- k=1: $|z_2| \geq \lambda |z_1|$
- k=2: $|z_3| \geq \lambda |z_2| \geq \lambda(\lambda |z_1|) = \lambda^2 |z_1|$
- ...
- k=n-1: $|z_n| \geq \lambda^{n-1} |z_1|$

Подставляя $|z_1| = |c|$, получаем:

$$|z_n| \geq \lambda^{n-1} |c|$$

Вывод

Поскольку λ — это константа, строго большая 1, и $|c|$ — положительное число, показатель λ^{n-1} стремится к бесконечности при $n \rightarrow \infty$:

$$\lim_{n \rightarrow \infty} |z_n| \geq \lim_{n \rightarrow \infty} |c| \cdot \lambda^{n-1}$$

Последовательность $\{|z_n|\}$ неограничена. Следовательно, $|c| > 2$ не принадлежит множеству Мандельброта.

Программа для изображения множества Мандельброта

```
import argparse
import numpy as np

import matplotlib.pyplot as plt
```

```

def mandelbrot(width = 900, height = 600, center=(-0.75, 0.0), zoom=1.0, ma
    cx, cy = center
    x_span = 3.0 / zoom
    y_span = 3.0 * height / width / zoom
    x_min, x_max = cx - x_span * 0.5, cx + x_span * 0.5
    y_min, y_max = cy - y_span * 0.5, cy + y_span * 0.5

    # сетка комплексных чисел
    xs = np.linspace(x_min, x_max, width, dtype=np.float64)
    ys = np.linspace(y_min, y_max, height, dtype=np.float64)
    X, Y = np.meshgrid(xs, ys)
    C = X + 1j * Y

    Z = np.zeros_like(C)

    it_counts = np.full(C.shape, maxiter, dtype=np.float64)
    mask = np.ones(C.shape, dtype=bool)

    for i in range(maxiter):
        Z[mask] = Z[mask] * Z[mask] + C[mask]
        escaped = np.abs(Z) > 2.0
        newly = escaped & mask
        if np.any(newly):
            zn = Z[newly]
            it_counts[newly] = i + 1 - np.log(np.log(np.abs(zn))) / np.log(
            mask &= ~escaped
            if not mask.any():
                break

    return it_counts, (x_min, x_max, y_min, y_max)

def plot_and_save(it_counts, extent, cmap="inferno", out=None, dpi=150, cen
    plt.figure(figsize=(it_counts.shape[1] / dpi, it_counts.shape[0] / dpi)

    im = plt.imshow(it_counts, cmap=cmap, extent=extent, origin='lower', in

    # Настраиваем оси
    ax = plt.gca()
    ax.set_xlabel('Re(c)', fontsize=10, color='white', weight='bold')
    ax.set_ylabel('Im(c)', fontsize=10, color='white', weight='bold')

    # Настраиваем цвет и стиль осей
    ax.spines['bottom'].set_color('white')

```

```

ax.spines['top'].set_color('white')
ax.spines['left'].set_color('white')
ax.spines['right'].set_color('white')
ax.spines['bottom'].set_linewidth(2)
ax.spines['top'].set_linewidth(2)
ax.spines['left'].set_linewidth(2)
ax.spines['right'].set_linewidth(2)

# Настраиваем деления и подписи
ax.tick_params(colors='white', which='both', labelsize=8)

plt.show()

```

```

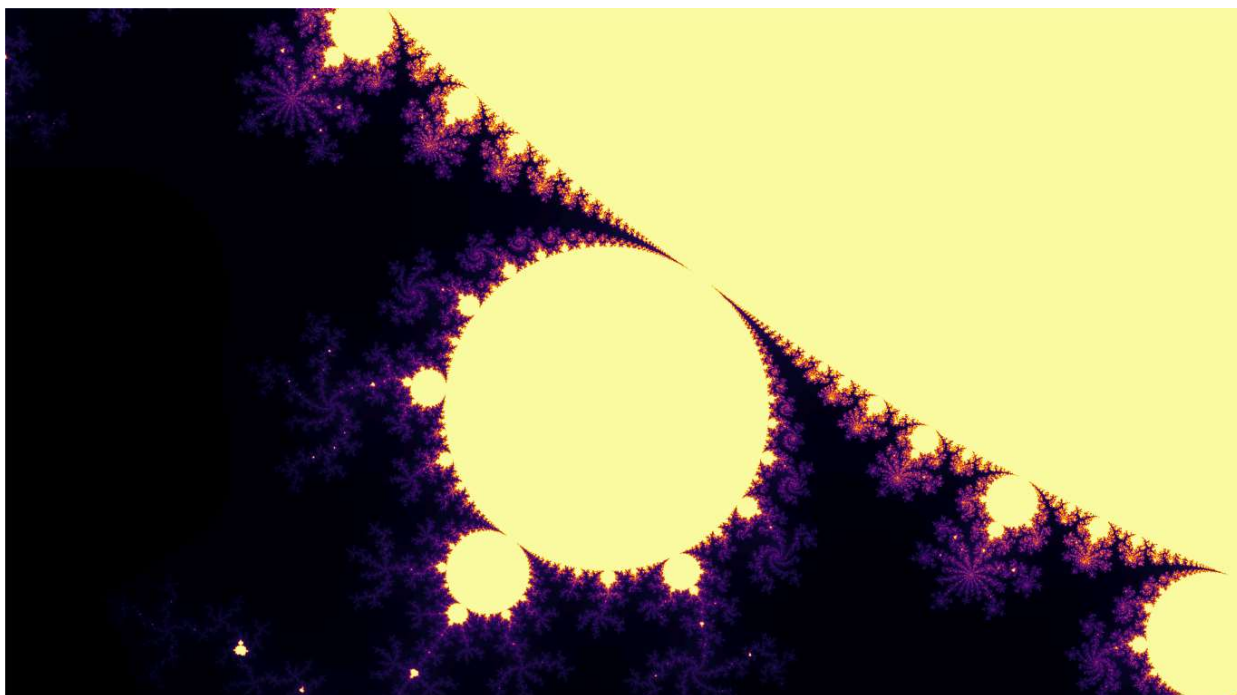
def main():
    parser = argparse.ArgumentParser()
    parser.add_argument('--width', type=int, default=1920)
    parser.add_argument('--height', type=int, default=1080)
    parser.add_argument('--maxiter', type=int, default=300)
    parser.add_argument('--center', type=float, nargs=2, default=[-0.75, 0.])
    parser.add_argument('--zoom', type=float, default=1.0)
    parser.add_argument('--cmap', type=str, default='magma')
    parser.add_argument('--out', type=str, default='mandelbrot.png')
    parser.add_argument('--dpi', type=int, default=150)

    args = parser.parse_args()

    it_counts, extent = mandelbrot(
        width=args.width,
        height=args.height,
        center=tuple(args.center),
        zoom=args.zoom,
        maxiter=args.maxiter
    )
    plot_and_save(it_counts, extent, cmap=args.cmap, out=args.out, dpi=args

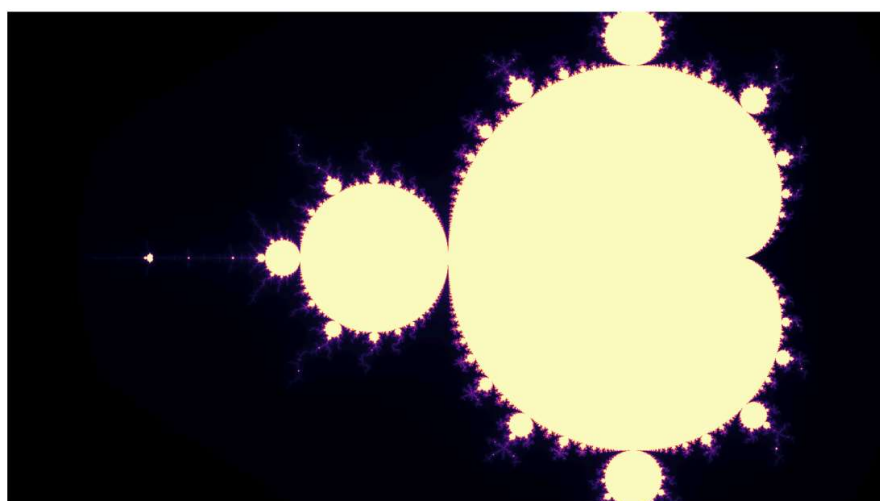
if __name__ == "__main__":
    main()

```



Итераций 500

Зум 10



Итераций 200

Зум 1