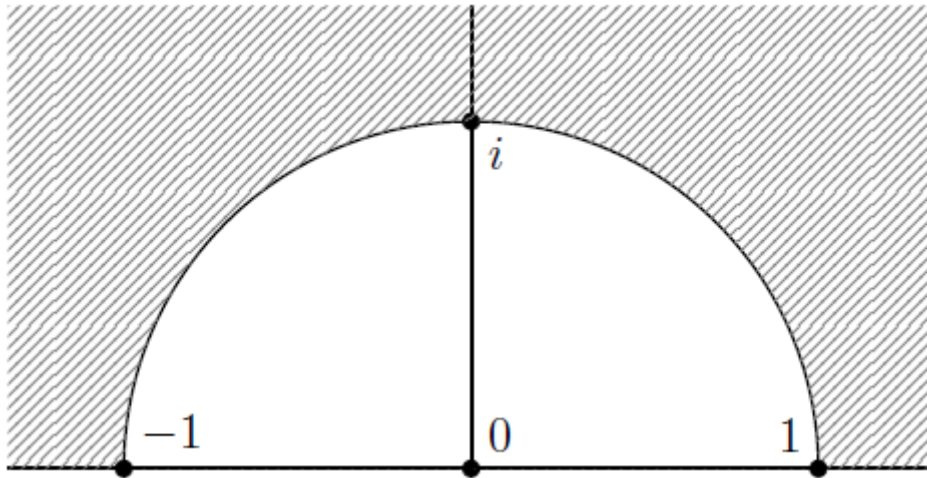


### Задание 1

Аналитически опишите заданные множества.



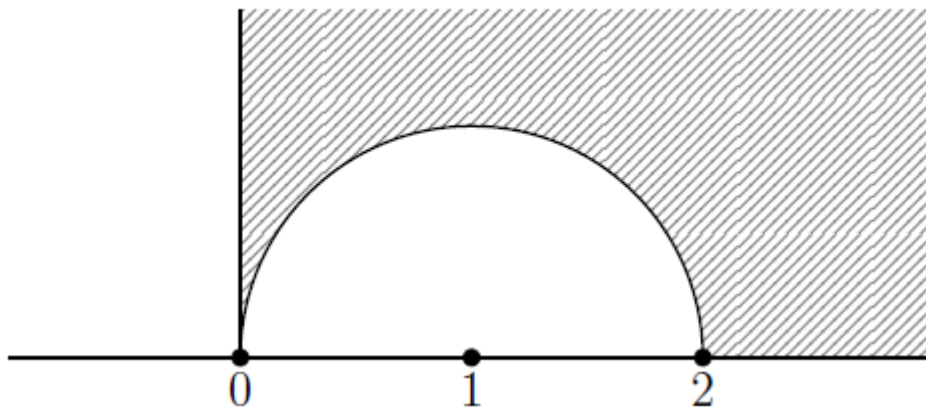
На графике заштрихована внешняя область относительно верхней полуокружности с центром в 0 и радиусом 1. Также вертикальная прямая через  $i$  ограничивает область сверху. Множество включает все точки вне окружности и выше действительной оси.

Аналитическое описание:

$$\{z \in \mathbb{C} \mid |z| \geq 1; \operatorname{Im}(z) \geq 0\}$$

Здесь:

- $|z| = \sqrt{\operatorname{Re}(z)^2 + \operatorname{Im}(z)^2}$  — действительная часть числа ( $z$ ),
- $\operatorname{Im}(z)$  — мнимая часть числа ( $z$ ).



На графике заштрихована внешняя область относительно верхней полуокружности с центром в точке 1 и радиусом 1. Эта область также ограничена слева вертикальной прямой через 0.

Аналитическое описание:

$$\{z \in \mathbb{C} \mid ((\operatorname{Re}(z) - 1)^2 + \operatorname{Im}(z)^2) \geq 1; \operatorname{Re}(z) \geq 0; \operatorname{Im}(z) \geq 0\}$$

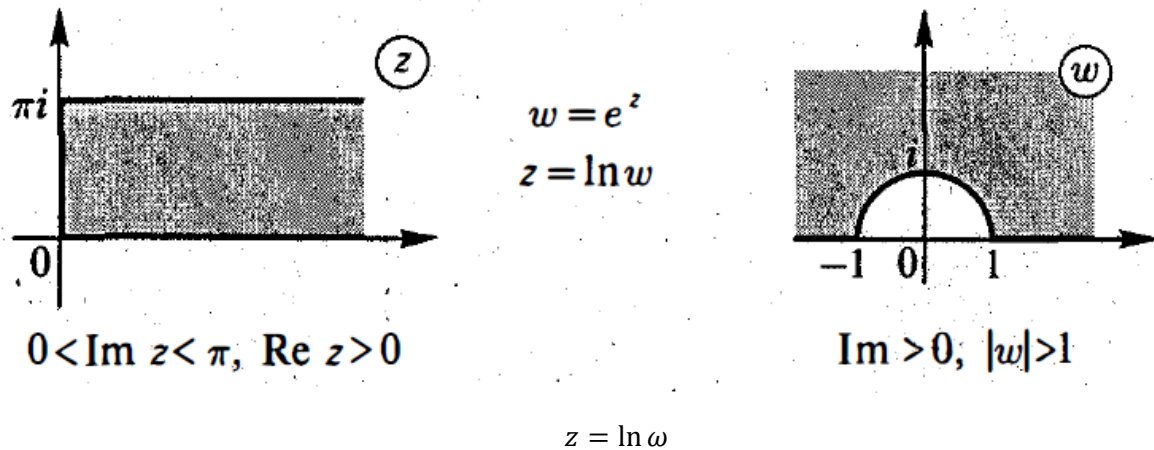
Здесь:

- $\operatorname{Re}(z)$  — действительная часть числа ( $z$ ),
- $\operatorname{Im}(z)$  — мнимая часть числа ( $z$ ).

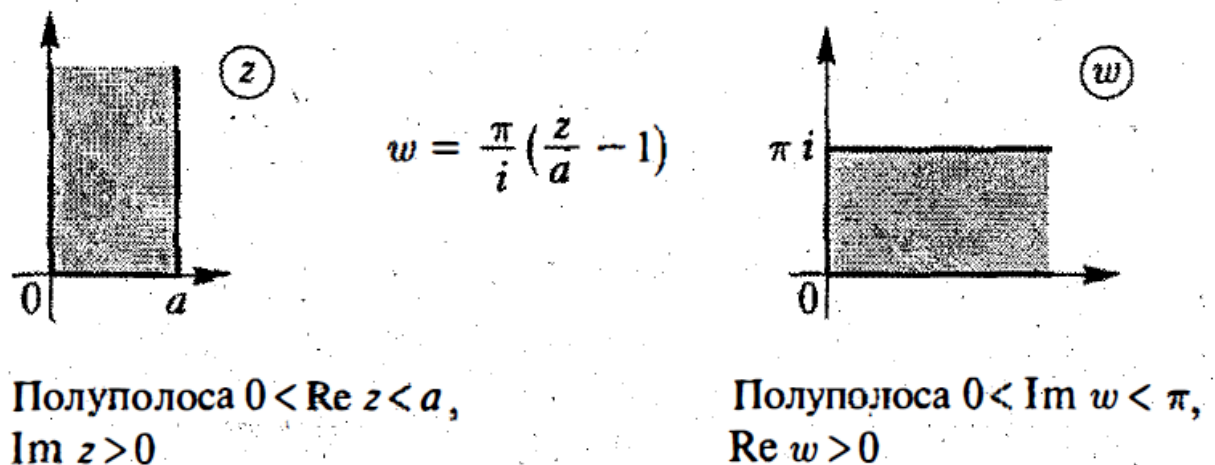
## Задание 2

Воспользовавшись композицией классических преобразований, составим конформное отображение, которое переводит первую область во вторую.

Для того, чтобы получить из 3 рисунка 11 нужно сначала выполнить данное преобразование:



Затем преобразование обратное данному при условии  $a = 1$ :



$$\omega = \frac{\pi}{i} \left( \frac{z}{a} - 1 \right), \quad a = 1$$

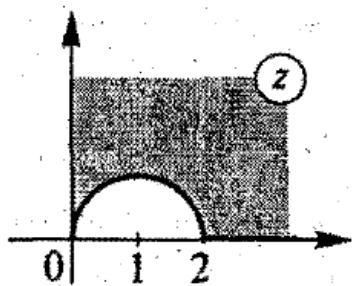
Для обратного преобразования выражаем  $z$ :

$$z = a \times \left( \frac{i}{\pi} \omega + 1 \right), \quad a = 1$$

Подставляя  $a = 1$  получаем:

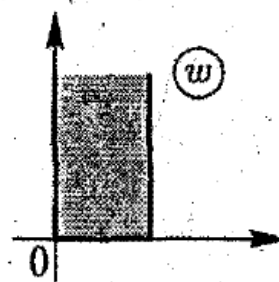
$$z = \left( \frac{i}{\pi} \omega + 1 \right)$$

И наконец преобразование обратное данному:



Угол  $\text{Im } z > 0, \text{Re } z > 0$   
с удаленным полукругом

$$w = -\frac{2}{z}$$



Полуполоса  $0 < \text{Re } w < 1, \text{Im } w > 0$

$$\omega = -\frac{2}{z}$$

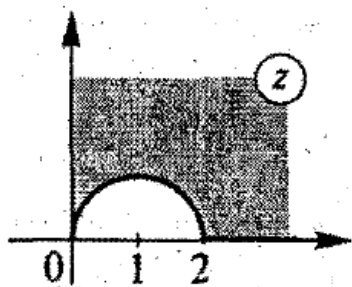
Получаем:

$$z = -\frac{2}{\omega}$$

### Задание 3

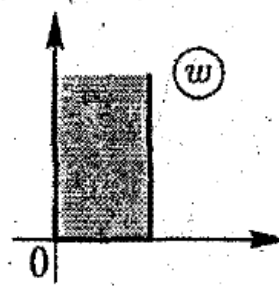
Составьте обратное отображение, переводящее второе множество в первое:

Для того, чтобы получить из 11 рисунка 3 нужно сначала выполнить данное преобразование:



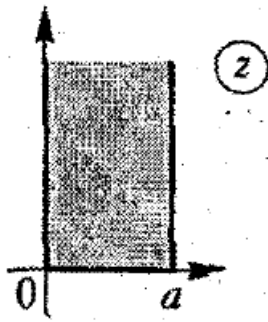
Угол  $\text{Im } z > 0, \text{Re } z > 0$   
с удаленным полукругом

$$w = -\frac{2}{z}$$

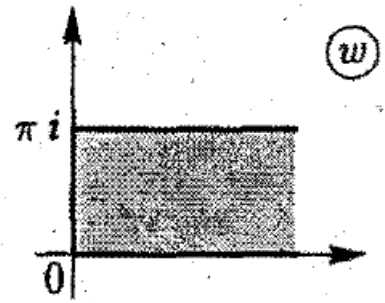


Полуполоса  $0 < \text{Re } w < 1, \text{Im } w > 0$

Затем преобразование при условии  $a = 1$ :



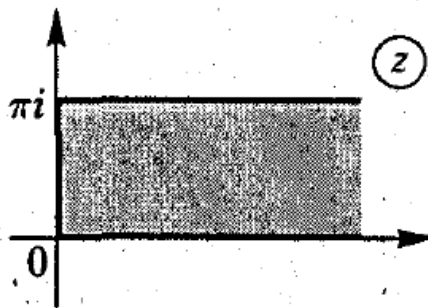
$$w = \frac{\pi}{i} \left( \frac{z}{a} - 1 \right)$$



Полуполоса  $0 < \operatorname{Re} z < a$ ,  
 $\operatorname{Im} z > 0$

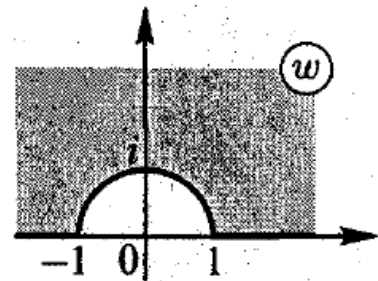
Полуполоса  $0 < \operatorname{Im} w < \pi$ ,  
 $\operatorname{Re} w > 0$

Выполним преобразование:



$$w = e^z$$

$$z = \ln w$$



$0 < \operatorname{Im} z < \pi$ ,  $\operatorname{Re} z > 0$

$\operatorname{Im} > 0$ ,  $|w| > 1$

$$\omega = e^z$$

Таким образом мы получаем исходную область.

Код

```
from math import pi
import numpy as np
import matplotlib.pyplot as plt
```

```
POINT_SIZE = 1
POINTS_NUMBER = 2000
X_MIN = -10
X_MAX = 10
Y_MIN = -10
Y_MAX = 10
X_VIEW_MIN = -2.5
X_VIEW_MAX = 2.5
Y_VIEW_MIN = -2.5
Y_VIEW_MAX = 2.5
```

```
x = np.linspace(X_MIN, X_MAX, POINTS_NUMBER)
y = np.linspace(Y_MIN, Y_MAX, POINTS_NUMBER)
X, Y = np.meshgrid(x, y)
area = X + 1j * Y
```

```

basic_area = area[(area.imag >= 0) & (np.abs(area) > 1)]

area_1 = np.log(basic_area)
area_1 = area_1[(area_1.imag >= 0) & (area_1.imag <= pi) & (area_1.real >=
0)]

area_2 = (((1j * area_1) / np.pi) + 1)
area_2 = area_2[(area_2.imag >= 0) & (area_2.real <= 1) & (area_2.real >= 0)]

area_3 = np.conj(2 / area_2)
area_3 = area_3[(((area_3.real - 1)**2 + area_3.imag**2) >= 1) & (area_3.real
>= 0) & (area_3.imag >= 0)]

fig, axs = plt.subplots(1, 4, figsize=(25, 5))

axs[0].scatter(basic_area.real, basic_area.imag, color='red', s=POINT_SIZE,
label='Этап 1: Исходное множество')
axs[0].set_xlim(X_VIEW_MIN, X_VIEW_MAX)
axs[0].set_ylim(Y_VIEW_MIN, Y_VIEW_MAX)

axs[1].scatter(area_1.real, area_1.imag, color='green', s=POINT_SIZE,
label='Этап 2:')
axs[1].set_xlim(X_VIEW_MIN, X_VIEW_MAX)
axs[1].set_ylim(Y_VIEW_MIN, Y_VIEW_MAX)

axs[2].scatter(area_3.real, area_3.imag, color='blue', s=POINT_SIZE,
label='Этап 3:')
axs[2].set_xlim(X_VIEW_MIN, X_VIEW_MAX)
axs[2].set_ylim(Y_VIEW_MIN, Y_VIEW_MAX)

axs[3].scatter(area_4.real, area_4.imag, color='purple', s=POINT_SIZE,
label='Этап 4:')
axs[3].set_xlim(X_VIEW_MIN, X_VIEW_MAX)
axs[3].set_ylim(Y_VIEW_MIN, Y_VIEW_MAX)

for ax in axs:
    ax.legend()
    ax.grid(True)
    ax.axhline(0, color='black', linewidth=0.5)
    ax.axvline(0, color='black', linewidth=0.5)
    ax.set_aspect('equal')

plt.tight_layout()
plt.show()

```

