1 Problem 1: Uniform Flow

Visualize the streamlines of a uniform flow along the x-axis. The stream function is:

$$\psi = Uy,$$

where $U = 1$.

```
import numpy as np
import matplotlib.pyplot as plt

# Grid
x = np.linspace(-5, 5, 100)
y = np.linspace(-5, 5, 100)
X, Y = np.meshgrid(x, y)

# Stream function
U = 1
psi = U * Y

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='viridis')
plt.title("Streamlines for Uniform Flow")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

2 Problem 2: Radial Source Flow

Visualize the streamlines for a source located at the origin. The stream function is:

$$\psi = \frac{m}{2\pi}\theta,$$

where $m > 0$.

```
x = np.linspace(-5, 5, 100)
y = np.linspace(-5, 5, 100)
```

```
X, Y = np.meshgrid(x, y)
r = np.sqrt(X**2 + Y**2)
theta = np.arctan2(Y, X)

# Stream function
m = 1  # Source strength
psi = (m / (2 * np.pi)) * theta

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='cool')
plt.title("Streamlines for a Source")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

3 Problem 3: Equipotential Lines for Uniform Flow

Visualize the equipotential lines for uniform flow along the x-axis. The velocity potential is:

$$\phi = Ux.$$

```
# Equipotential function
phi = U * X  # Velocity potential for uniform flow

# Plot equipotential lines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, phi, levels=30, cmap='coolwarm')
plt.title("Equipotential Lines for Uniform Flow")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

4 Problem 4: Flow Due to a Sink

Visualize the streamlines for a sink located at the origin in 2D flow. The stream function is:

$$\psi = -\frac{m}{2\pi}\theta,$$

where $m > 0$.

```
# Stream function for a sink
psi = -(m / (2 * np.pi)) * theta

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='plasma')
plt.title("Streamlines for a Sink")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

5 Problem 5: Flow Around a Doublet

Simulate the flow field for a doublet at the origin. The stream and equipotential functions for the doublet are:

$$\psi = -\frac{\mu \sin\theta}{2\pi r}, \quad \phi = -\frac{\mu \cos\theta}{2\pi r}.$$

```
mu = 10  # Doublet strength

# Stream function
psi = -(mu / (2 * np.pi)) * np.sin(theta) / r

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='viridis')
plt.title("Streamlines for a Doublet")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
```

```
plt.grid()
plt.show()
```

—

6 Problem 6: Flow Around a Cylinder

Visualize the flow field around a stationary cylinder in a uniform flow. The stream function is:

$$\psi = Ur\sin\theta\left(1 - \frac{a^2}{r^2}\right),$$

where $a = 1$ is the cylinder radius.

```
a = 1  # Cylinder radius
psi = U * r * np.sin(theta) * (1 - (a**2 / r**2))

# Mask out the inside of the cylinder
psi[r < a] = np.nan

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='coolwarm')
plt.title("Streamlines Around a Circular Cylinder")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

7 Problem 7: Stagnation Point Flow

Simulate stagnation point flow in 2D. The stream function is:

$$\psi = \frac{a}{2}(x^2 - y^2).$$

```
a = 1  # Flow strength
psi = 0.5 * a * (X**2 - Y**2)
```

```
# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='plasma')
plt.title("Stagnation Point Flow")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

8 Problem 8: Combined Source and Uniform Flow

Simulate the flow field resulting from the superposition of a uniform flow and a source. The stream function is:

$$\psi = Uy + \frac{m}{2\pi}\theta.$$

```
psi = U * Y + (m / (2 * np.pi)) * theta

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='viridis')
plt.title("Streamlines for Uniform Flow with a Source")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```

—

9 Problem 9: Pathlines in a Simple Flow

Determine the pathlines of a fluid particle in a velocity field defined by:

$$u = x, \quad v = -y.$$

```
# Velocity field
u = X
v = -Y

# Initial condition
x0, y0 = 1, 1

# Pathline simulation
dt = 0.01
steps = 100
x_path = [x0]
y_path = [y0]

for _ in range(steps):
    x = x_path[-1]
    y = y_path[-1]
    x_path.append(x + u[int(x*10), int(y*10)] * dt)
    y_path.append(y + v[int(x*10), int(y*10)] * dt)

# Plot pathline
plt.figure(figsize=(6, 6))
plt.plot(x_path, y_path, label="Pathline")
plt.title("Pathline of a Particle")
plt.xlabel("x")
plt.ylabel("y")
plt.grid()
plt.legend()
plt.show()
```

—

10 Problem 10: Streamlines for a Source and Sink Pair

Simulate the streamlines for a source and sink pair located symmetrically about the origin. The stream function is:

$$\psi = \frac{m}{2\pi} \ln \frac{r_1}{r_2},$$

```
# Source and sink at (+1, 0) and (-1, 0)
psi_source = (m / (2 * np.pi)) * np.arctan2(Y, X - 1)
```

```python
psi_sink = -(m / (2 * np.pi)) * np.arctan2(Y, X + 1)

# Total stream function
psi = psi_source + psi_sink

# Plot streamlines
plt.figure(figsize=(6, 6))
plt.contour(X, Y, psi, levels=30, cmap='cool')
plt.title("Streamlines for a Source and Sink Pair")
plt.xlabel("x")
plt.ylabel("y")
plt.axis("equal")
plt.grid()
plt.show()
```