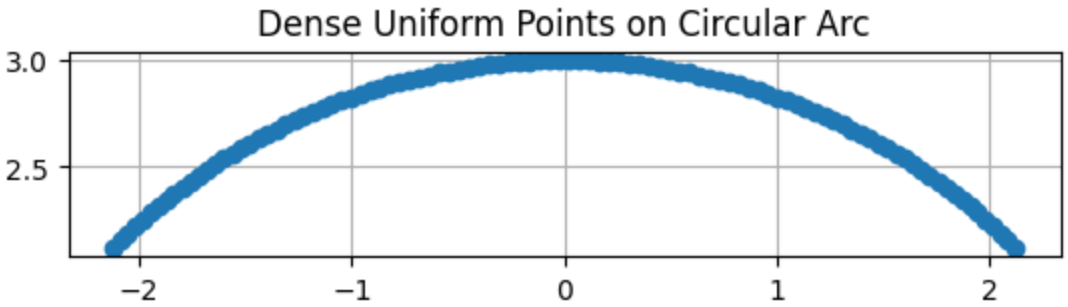
Amit Divekar | Practical 10

Uniform Points on Circular Arcs

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

Q1. Generate and plot a large number of uniformly spaced points on a circular arc to show smoothness.

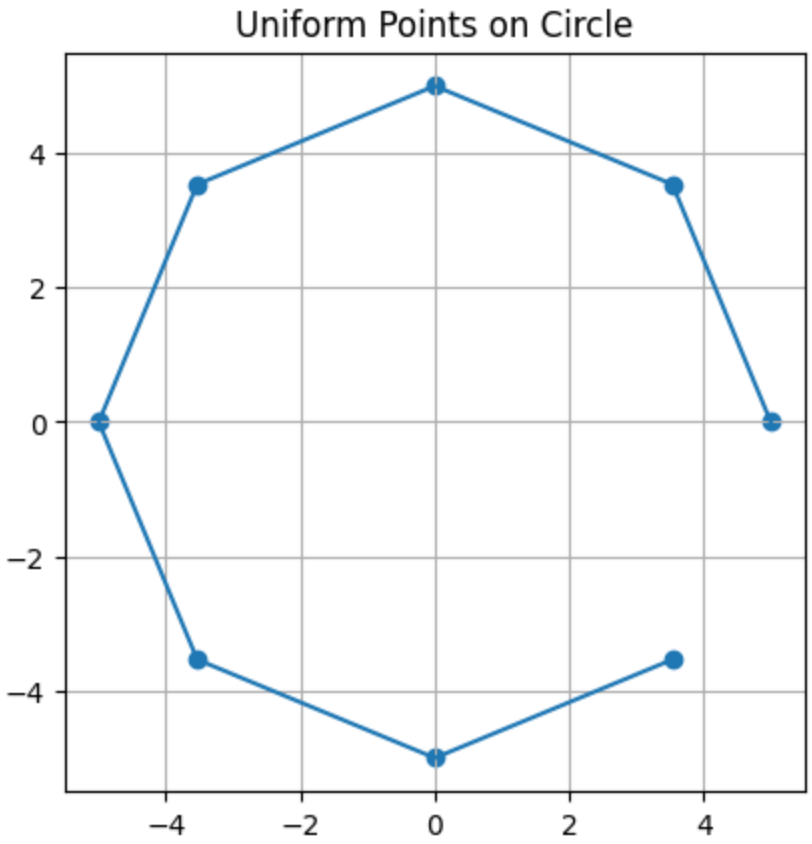
In [2]:
```
import numpy as np
import matplotlib.pyplot as plt
r=3
n = 100
theta = np.linspace(np.pi/4, 3*np.pi/4, n)
x = r * np.cos(theta)
y = r * np.sin(theta)
plt.plot(x, y, 'o')
plt.gca().set_aspect('equal')
plt.title("Dense Uniform Points on Circular Arc")
plt.grid()
plt.show()
```



Q2. Generate and plot n uniformly spaced points on the circumference of a circle $x^2+y^2=r^2$ with radius r=5

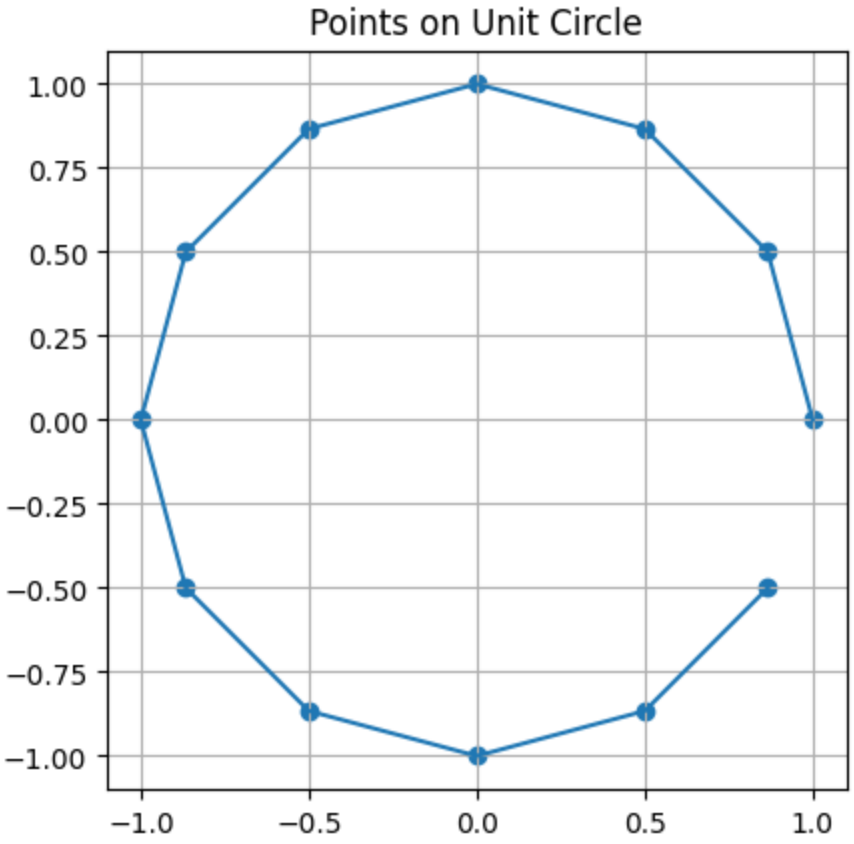
In [3]:
```
import numpy as np
import matplotlib.pyplot as plt
r=5
n=8
theta = np.linspace(0, 2*np.pi, n, endpoint=False)
x = r * np.cos(theta)
y = r * np.sin(theta)
plt.scatter(x, y)
```

```
plt.plot(x, y)
plt.gca().set_aspect('equal')
plt.title("Uniform Points on Circle")
plt.grid()
plt.show()
```



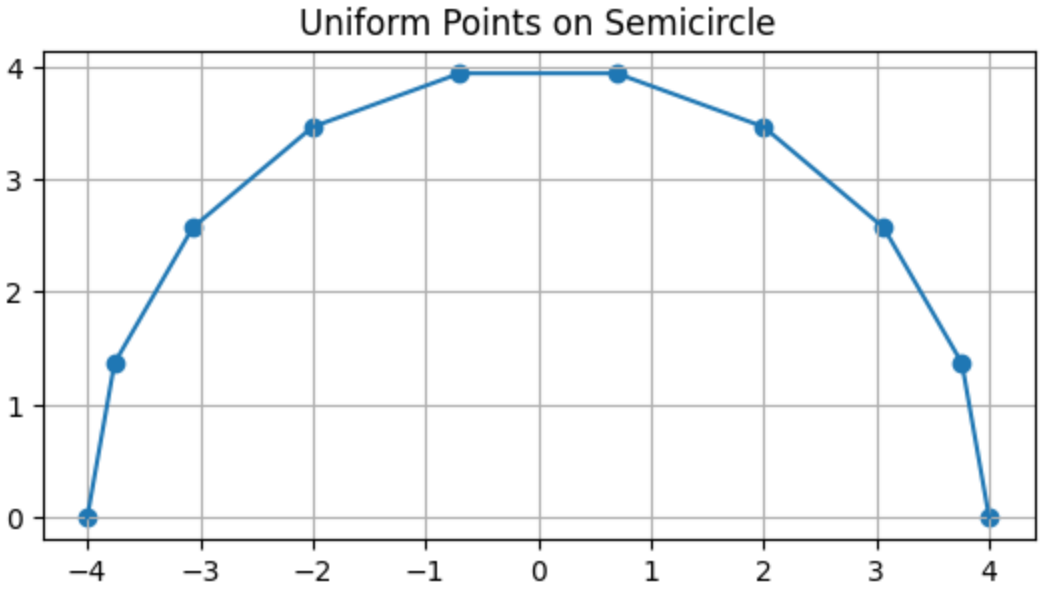
Q3. Generate and plot 12 uniformly spaced points on a unit circle.

In [4]:
```
import numpy as np
import matplotlib.pyplot as plt
n = 12
theta = np.linspace(0, 2*np.pi, n, endpoint=False)
x = np.cos(theta)
y = np.sin(theta)
plt.scatter(x, y)
plt.plot(x, y)
plt.gca().set_aspect('equal')
plt.title("Points on Unit Circle")
plt.grid()
plt.show()
```



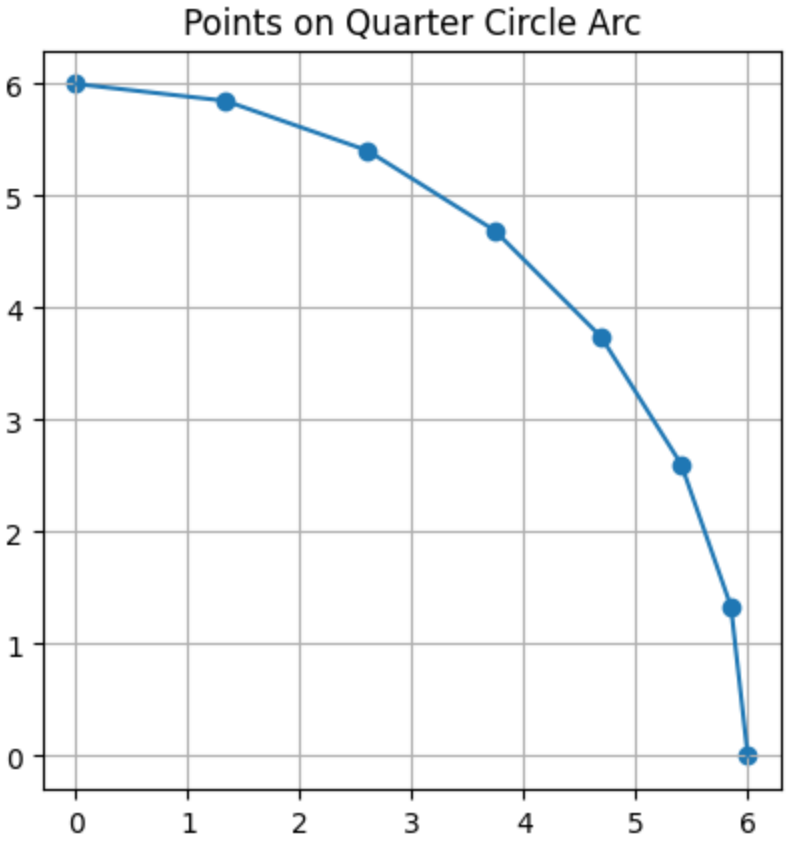
Q4. Generate and plot n uniformly spaced points on the upper semicircle of radius 4.

In [5]:
```
import numpy as np
import matplotlib.pyplot as plt
r=4
n = 10
theta = np.linspace(0, np.pi, n)
x = r * np.cos(theta)
y = r * np.sin(theta)
plt.scatter(x, y)
plt.plot(x, y)
plt.gca().set_aspect('equal')
plt.title("Uniform Points on Semicircle")
plt.grid()
plt.show()
```



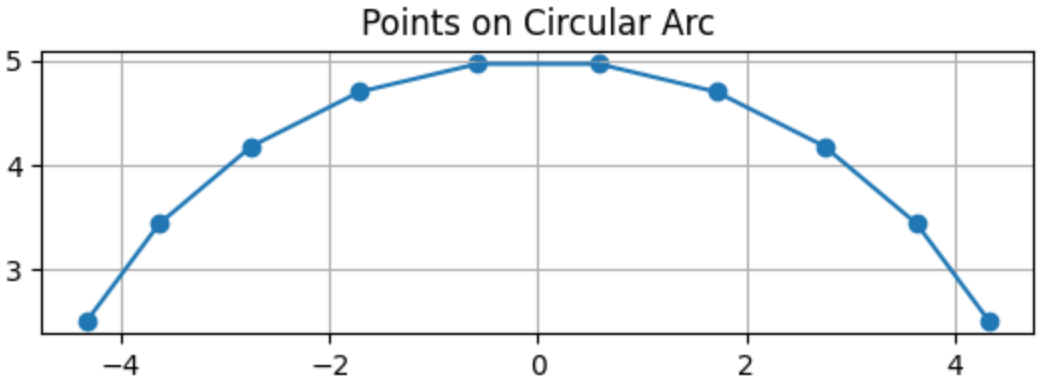
Q5. Generate and plot uniformly spaced points on a quarter circle arc of radius 6.

In [6]:
```
import numpy as np
import matplotlib.pyplot as plt
r=6
n=8
theta = np.linspace(0, np.pi/2, n)
x = r * np.cos(theta)
y = r * np.sin(theta)
plt.scatter(x, y)
plt.plot(x, y)
plt.gca().set_aspect('equal')
plt.title("Points on Quarter Circle Arc")
plt.grid()
plt.show()
```



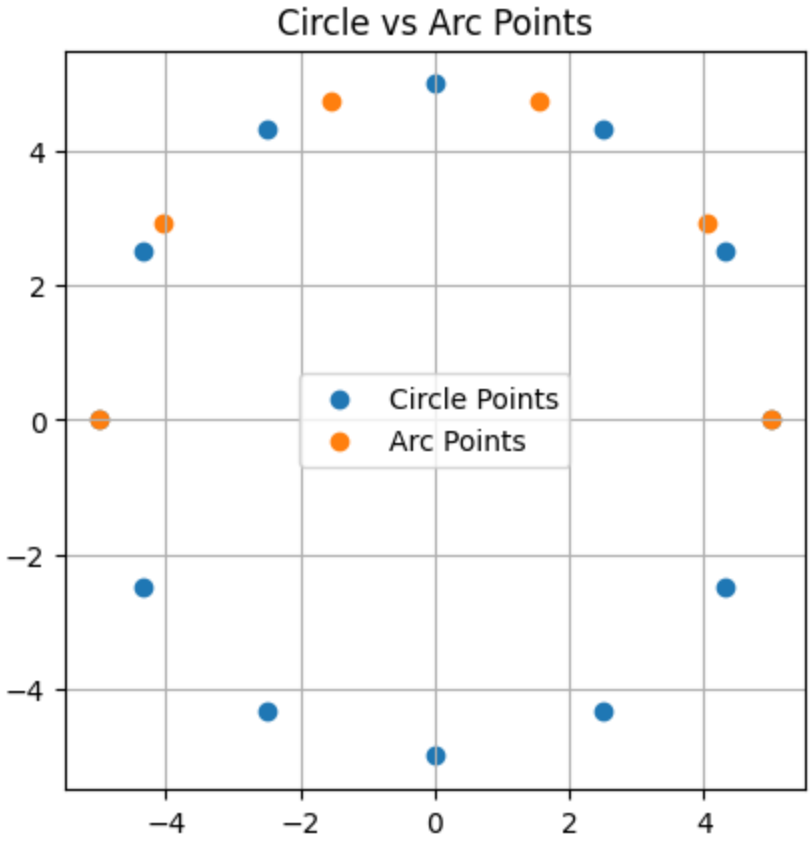
Q6. Generate uniformly spaced points on a circular arc from 30° to 150° with radius 5.

In [7]:
```
import numpy as np
import matplotlib.pyplot as plt
r=5
n = 10
theta = np.linspace(np.pi/6, 5*np.pi/6, n)
x = r * np.cos(theta)
y = r * np.sin(theta)
plt.scatter(x, y)
plt.plot(x, y)
plt.gca().set_aspect('equal')
plt.title("Points on Circular Arc")
plt.grid()
plt.show()
```



Q7. Plot uniform points on a full circle and points on a semicircle arc together.

In [8]:
```
import numpy as np
import matplotlib.pyplot as plt
r=5
theta1 = np.linspace(0, 2*np.pi, 12, endpoint=False)
theta2 = np.linspace(0, np.pi, 6)
x1, y1 = r*np.cos(theta1), r*np.sin(theta1)
x2, y2 = r*np.cos(theta2), r*np.sin(theta2)
plt.scatter(x1, y1, label="Circle Points")
plt.scatter(x2, y2, label="Arc Points")
plt.gca().set_aspect('equal')
plt.legend()
plt.title("Circle vs Arc Points")
plt.grid()
plt.show()
```


Circle vs Arc Points
Circle Points
Arc Points
4
2
0
−2
−4
−4
−2
0
2
4