

# Bresenham Circle Drawing Algorithm

April 2, 2025

```
[1]: import matplotlib.pyplot as plt
```

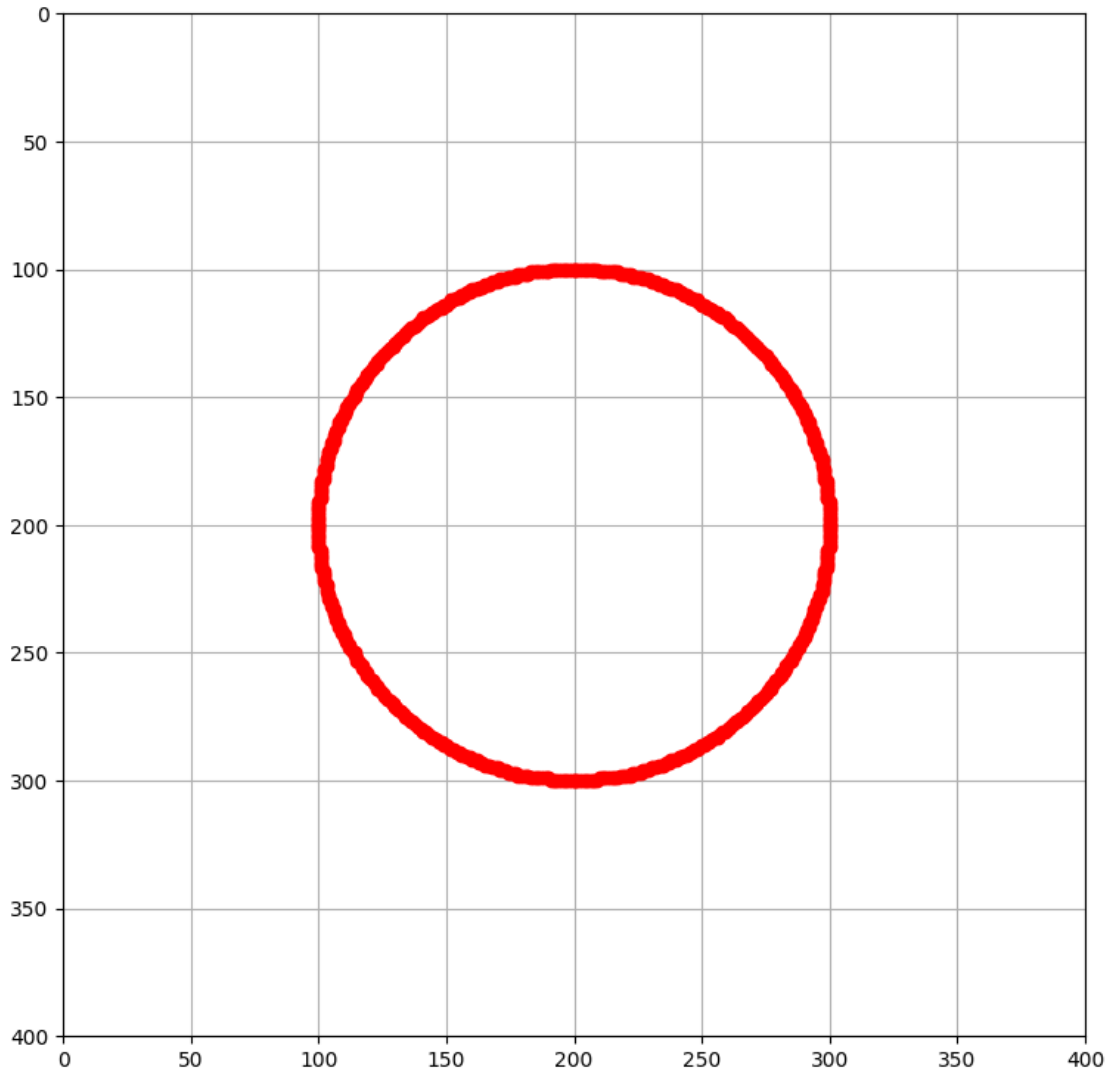
```
[2]: def drawCircle(xc, yc, x, y, ax):  
    ax.plot(xc + x, yc + y, 'ro')  
    ax.plot(xc - x, yc + y, 'ro')  
    ax.plot(xc + x, yc - y, 'ro')  
    ax.plot(xc - x, yc - y, 'ro')  
    ax.plot(xc + y, yc + x, 'ro')  
    ax.plot(xc - y, yc + x, 'ro')  
    ax.plot(xc + y, yc - x, 'ro')  
    ax.plot(xc - y, yc - x, 'ro')  
  
    def circleBres(xc, yc, r, ax):  
        x = 0  
        y = r  
        d = 3 - 2 * r  
        drawCircle(xc, yc, x, y, ax)  
  
        while y >= x:  
            if d > 0:  
                y -= 1  
                d = d + 4 * (x - y) + 10  
            else:  
                d = d + 4 * x + 6  
  
            x += 1  
            drawCircle(xc, yc, x, y, ax)
```

```
[4]: def main():  
    xc, yc = 200, 200  
    r = 100  
  
    fig, ax = plt.subplots(figsize=(9, 9))  
    ax.set_aspect('equal')  
  
    ax.set_xlim(0, 400)  
    ax.set_ylim(0, 400)
```

```
circleBres(xc, yc, r, ax)

plt.gca().invert_yaxis()
plt.grid()
plt.show()

if __name__ == "__main__":
    main()
```



[ ]: