**SOURCE CODE**

1. # to implement mid point circle algorithm
2. **import** pygame
3. **import** sys
4. **def** mid\_point\_circle(xc, yc, r):
5. x **=** 0
6. y **=** r
7. p **=** 1 **-** r
9. points **=** []
11. **while** x <**=** y:
12. points.append((xc **+** x, yc **+** y))
13. points.append((xc **-** x, yc **+** y))
14. points.append((xc **+** x, yc **-** y))
15. points.append((xc **-** x, yc **-** y))
16. points.append((xc **+** y, yc **+** x))
17. points.append((xc **-** y, yc **+** x))
18. points.append((xc **+** y, yc **-** x))
19. points.append((xc **-** y, yc **-** x))
21. **if** p < 0:
22. p **+=** 2 **\*** x **+** 3
23. **else**:
24. p **+=** 2 **\*** (x **-** y) **+** 5
25. y **-=** 1
26. x **+=** 1
28. **return** points
30. pygame.init()
31. WIDTH **=** 800
32. HEIGHT **=** 600
33. screen **=** pygame.display.set\_mode((WIDTH, HEIGHT))
34. pygame.display.set\_caption("Midpoint Circle Algorithm")
35. WHITE **=** (255, 255, 255)
36. BLACK **=** (0, 0, 0)
37. screen.fill(BLACK)
38. xc, yc, r **=** WIDTH **//** 2, HEIGHT **//** 2, 100
39. circle\_points **=** mid\_point\_circle(xc, yc, r)
40. **for** point **in** circle\_points:
41. screen.set\_at(point, WHITE)
42. pygame.display.flip()
43. **while** True:
44. **for** event **in** pygame.event.get():
45. **if** event.type **==** pygame.QUIT:
46. pygame.quit()
47. sys.exit()
48. pygame.time.delay(100)

**OUTPUT**

