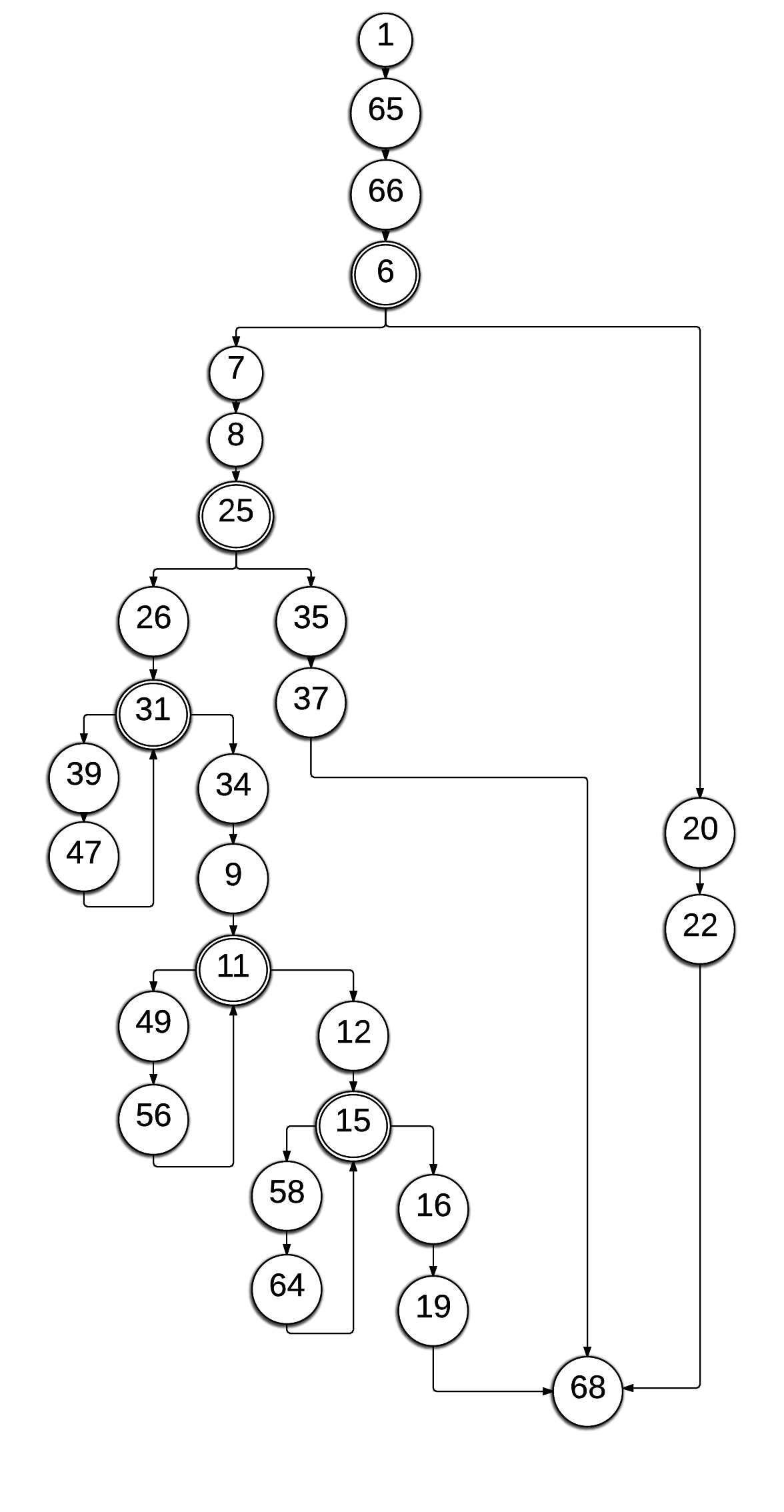
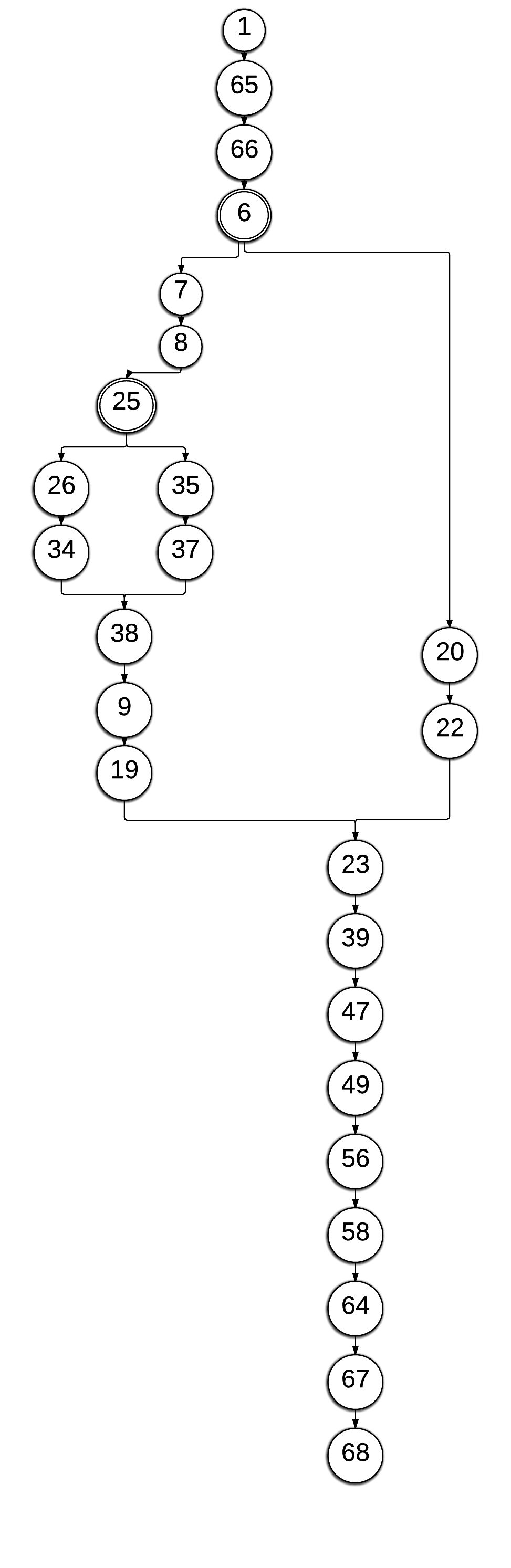
1. **public class** AES {
2. **byte**[] **skey** = **new byte**[1000];
3. String **skeyString**;
4. **static byte**[] *raw*;
5. String **inputMessage**,**encryptedData**,**decryptedMessage**;
6. **public** AES() {
7. **try** {
8. generateSymmetricKey();
9. **inputMessage**=JOptionPane.showInputDialog(**null**,**"Enter message to  
    encrypt"**);
10. **byte**[] ibyte = **inputMessage**.getBytes();
11. **byte**[] ebyte=*encrypt*(*raw*, ibyte);
12. String encryptedData = **new** String(ebyte);
13. System.***out***.println(**"Encrypted message "**+encryptedData);
14. JOptionPane.showMessageDialog(**null**,**"Encrypted  
     Data"**+**"\n"**+encryptedData);
15. **byte**[] dbyte= *decrypt*(*raw*,ebyte);
16. String decryptedMessage = **new** String(dbyte
17. System.***out***.println(**"Decrypted message** +decryptedMessage
18. JOptionPane.showMessageDialog(**null**,**"Decrypted Data  
     "**+**"\n"**+decryptedMessage);
19. }
20. **catch**(Exception e) {
21. System.***out***.println(e);
22. }
23. }
24. *//Symmetric key generation*
25. **void** generateSymmetricKey() {
26. **try** {
27. Random r = **new** Random();
28. **int** num = r.nextInt(10000);
29. String knum = String.*valueOf*(num);
30. **byte**[] knumb = knum.getBytes();
31. **skey**=*getRawKey*(knumb);
32. **skeyString** = **new** String(**skey**);
33. System.***out***.println(**"AES Symmetric key = "**+**skeyString**);
34. }
35. **catch**(Exception e) {
36. System.***out***.println(e);
37. }
38. }
39. **private static byte**[] getRawKey(**byte**[] seed) **throws** Exception {
40. KeyGenerator kgen = KeyGenerator.*getInstance*(**"AES"**);
41. SecureRandom sr = SecureRandom.*getInstance*(**"SHA1PRNG"**);
42. sr.setSeed(seed);
43. kgen.init(128, sr);
44. SecretKey skey = kgen.generateKey();
45. *raw* = skey.getEncoded();
46. **return** *raw*;
47. }
48. *//encrypt portion*
49. **private static byte**[] encrypt(**byte**[] raw, **byte**[] clear) **throws** Exception
50. {
51. SecretKeySpec skeySpec = **new** SecretKeySpec(raw, **"AES"**);
52. Cipher cipher = Cipher.*getInstance*(**"AES"**);
53. cipher.init(Cipher.***ENCRYPT\_MODE***, skeySpec);
54. **byte**[] encrypted = cipher.doFinal(clear);
55. **return** encrypted;
56. }
57. *//decrypt portion*
58. **private static byte**[] decrypt(**byte**[] raw, **byte**[] encrypted) **throws** Exception {
59. SecretKeySpec skeySpec = **new** SecretKeySpec(raw, **"AES"**);
60. Cipher cipher = Cipher.*getInstance*(**"AES"**);
61. cipher.init(Cipher.***DECRYPT\_MODE***, skeySpec);
62. **byte**[] decrypted = cipher.doFinal(encrypted);
63. **return** decrypted;
64. }
65. **public static void** main(String args[]) {
66. AES aes = **new** AES();
67. }
68. }

Node =25

Edge = 26





Node = 27

Edge = 31