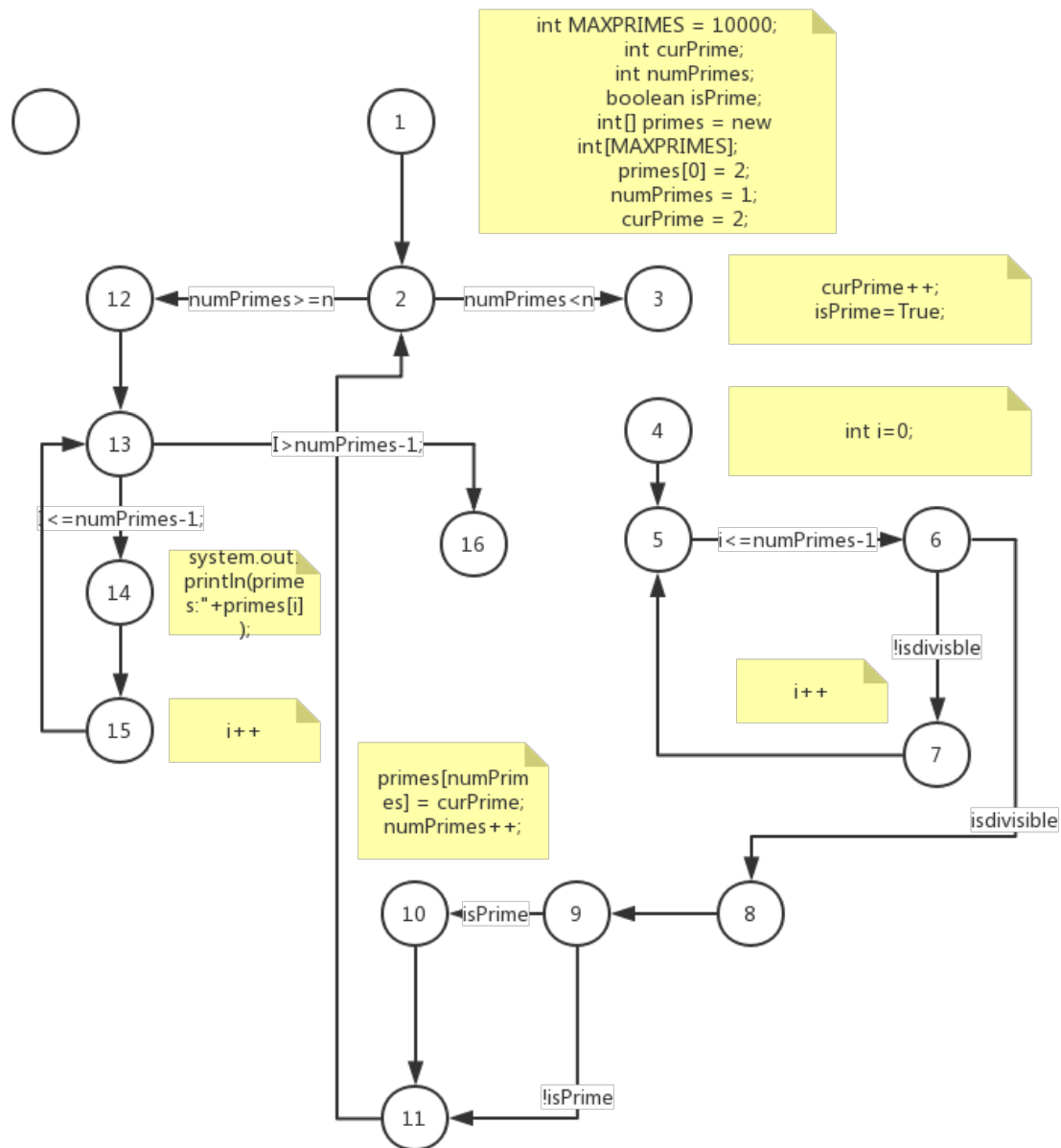


作业3

流程图



3 Consider test cases $t_1 = (n = 3)$ and $t_2 = (n = 5)$. Although these tour the same prime paths in `printPrime()`, they don't necessarily find the same faults. Design a simple fault that t_2 would be more likely to discover than t_1 would.

可能是数组越界错误，如 `MAXPRIMES = 4`

4. For `printPrime()`, find a test case such that the corresponding test path visits the edge that connects the beginning of the while statement to the for statement without going through the body of the while loop.

例如 $n=1$ 的时候就不会进入循环

5. Enumerate the test requirements for node coverage, edge coverage, and prime path coverage for the path for `printPrimes()`.

点覆盖: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15}

边覆盖: { (1, 2) , (2, 3) , (3, 4) , (3, 12) , (4, 5) , (5, 6) , (6, 9) , (6, 7) , (7, 8) , (7, 10) , (8, 9) , (9, 11) , (9, 3) , (10, 6) , (11, 3) , (12, 13) , (13, 14) , (14, 15) , (15, 13) , (13, 16) }

主路径覆盖: { (1, 2, 3, 4, 5, 6, 7, 8, 9, 11) , (1, 2, 3, 4, 5, 6, 7, 10) , (1, 2, 3, 12, 13, 16) , (1, 2, 3, 12, 13, 14, 15) , (3, 4, 5, 6, 7, 8, 9, 11, 3) , (4, 5, 6, 7, 8, 9, 11, 3, 4) , (5, 6, 7, 8, 9, 11, 3, 4, 5) , (6, 7, 10, 6) , (6, 7, 8, 9, 11, 3, 4, 5, 6) , (7, 10, 6, 7) , (7, 8, 9, 11, 3, 4, 5, 6, 7) , (8, 9, 11, 3, 4, 5, 6, 7, 8) , (9, 11, 3, 4, 5, 6, 7, 8, 9) , (10, 6, 7, 10) , (11, 3, 4, 5, 6, 7, 8, 9, 11) , (13, 14, 15, 13) , (14, 15, 13, 14) , (15, 13, 14, 15) , (14, 15, 13, 16) , (15, 13, 16) }

6. List test paths that achieve node coverage but not edge coverage on the graph.

{1,2,3,4,5,6,7,5,8,9,10,11,2,12,13,14,15,13,16}

7. List test paths that achieve edge coverage but not prime path coverage on the graph.

{ [1,2,3,4,5,6,7,5,8,9,10,11,2,13,14,15,13,16], [1,2,3,4,5,9,10,11,2,12,13,16] }



8. 测试代码

```
public class homework3Test {
    ByteArrayOutputStream out;
    PrintStream ps;
    public homework3 primes = new homework3();
    int [] print = {2, 3, 5};

    @Before
    public void setUp(){
        out = new ByteArrayOutputStream();
        ps = new PrintStream(out);
        System.setOut(ps);
    }

    @Test
    public void test() throws IOException {
        out.flush();
        homework3.printPrimes(3);
        assertEquals("Prime: 2\r\nPrime: 3\r\nPrime: 5\r\n", out.toString());
    }
}
```

9. 测试结果

| | | | | | |
|----------------------|---|--------|----|----|----|
| ▶ homework3.java |  | 67.1 % | 49 | 24 | 73 |
| ▶ homework3Test.java |  | 84.3 % | 43 | 8 | 51 |

