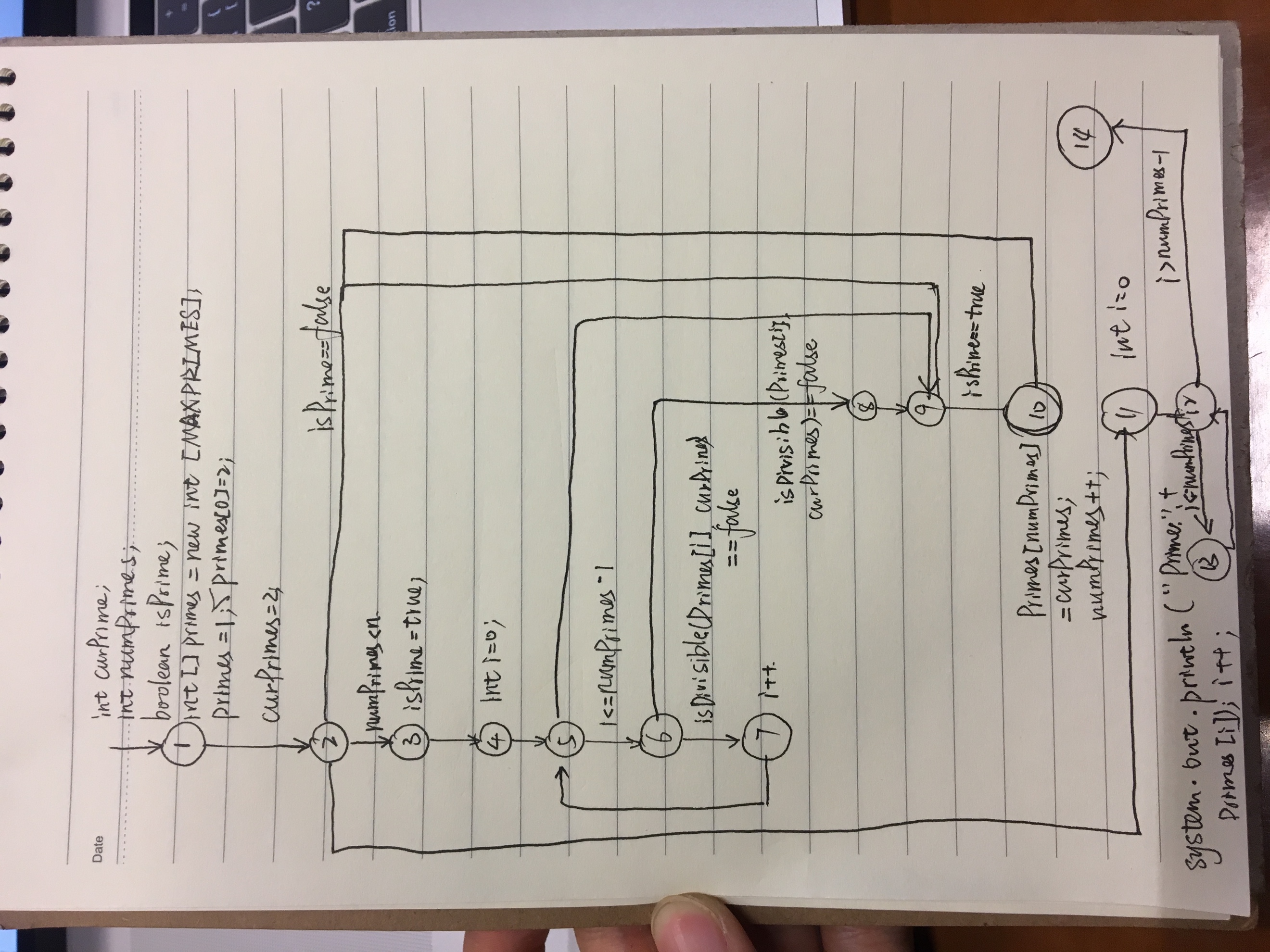
Exercise Section 2.3 • 7. Use the following method printPrimes() for questions a–d.

(a)



(b)将while 的循环判断条件修改为MAXPRIMES=4

(c)令n=1

(d)节点覆盖： TR={1,2,3,4,5,6,7,8,9,10,11,12,13,14}

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

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

基于Junit及Eclemma（jacoco）实现一个主路径覆盖的测试

package printPrime;

import static org.junit.Assert.\*;

import java.io.ByteArrayOutputStream;

import java.io.PrintStream;

import org.junit.Before;

import org.junit.Test;

public class primeTest {

prime p;

ByteArrayOutputStream str;

@Before

public void setup() throws Exception{

p = new prime();

str = new ByteArrayOutputStream();

System.setOut(new PrintStream(str));

}

@Test

public void test() {

String output = new String("Prime: 2\r\nPrime: 3\r\nPrime: 5\r\nPrime: 7\r\nPrime: 11\r\n");

p.printPrimes(5);

assertEquals(output, str.toString());

}

}

