Section 5.1 Introduction to Number Theory

2. When n = 47, the algorithm would check if the numbers 2, 3, 4, 5, and 6 (round down √47 to the nearest integer) are divisors of 47.
None of these integers divide 47, and the algorithm would determine that 47 is prime and return 0.

3. When n = 209 a quick computation shows that the algorithm would check the integers 2, 3, 4, …, 14 in order to find any possible divisors. When it checks 11, the algorithm will find that 11 does divide 209 and return the number 11.

12. gcd(0,17) = 17 According to the definition in the book any non-zero number is a divisor of 0, so the greatest divisor common to both is 17.

14. gcd(60,90) = 30

15 $110 = 2 \cdot 5 \cdot 11$ and $273 = 3 \cdot 7 \cdot 13$ so gcd(110,273) = 1

17. $315 = 3^2 \cdot 5 \cdot 7$ and $825 = 3 \cdot 5^2 \cdot 11$ so gcd(315,825)= (3)(5)=15

22. $\gcd(15,15^9) = 15$ (This follows from theorem 5.1.25 for instance.

25. $$\begin{aligned} &lcm(0,17) = 0 \quad lcm(60,90) = 180 \quad lcm(110,273) = 2 \cdot 3 \cdot 5 \cdot 7 \cdot 11 \cdot 13 = 30030 \\ &lcm(315,825) = 3^2 \cdot 5^2 \cdot 7 \cdot 11 = 17{,}325 \quad lcm(15,15^9) = 15^9 \end{aligned}$$

29. $$\begin{aligned} &\text{If } d_1 \mid m \text{ and } d_2 \mid n \text{ then } m = p \cdot d_1 \text{ and } n = q \cdot d_2 \text{ for some integers p and q} \\ &\text{but then } mn = pq \cdot d_1 d_2 \text{ and hence } d_1 d_2 \mid mn. \end{aligned}$$

30. $$\begin{aligned} &\text{If } dc \mid nc, \text{ then } nc = q \cdot dc. \\ &\text{This implies that } n = qd, \text{ hence } d \mid n \end{aligned}$$

Section 5.2 Representation of Integers and Integer Algorithms.

1. 60 (decimal) = 111100 (base 2), so it takes 6 bits to represent this number.

2. 63 (decimal) = 111111 (base 2), so it takes 6 bits.

5. 128 (decimal) = 10000000 (base 2). So it takes 8 bits.

8. 1001 -> 9

9. 11011 -> 27

10. 11011011 -> 219

14. 34 -> 100010

16. 223 -> 11011111

17. 400 -> 110010000

20. 1001 + 1111 = 11000

22. 110110 + 101101 = 1100011

23. 101101 + 11011 = 1001000

26. 3A -> 58

27. 1E9 -> 489

28. 3E7C -> 15,996

35. 4A + B4 = FE (hex addition)

36. 195 + 76E = 903 (hex addition)

40. 2010 can represent a number in decimal (the regular number 2,010) or hexadecimal (this would represent the decimal number 8,208), but not in binary (the latter does not allow for a 2)

41. 1101010 can represent a number in binary, decimal and hexadecimal.

56. See the back of the book.

57. Use the explanation in example 5.2.15 as an example.
If n = 15 we trace through the algorithm updating result = result*x every time n is odd. We update x to be x*x every time we execute a loop.

x	Current value of n	N mod 2	Result = result*x	New n
a	15	1	a	7
a^2	7	1	a^3	3
a^4	3	1	a^7	1
a^8	1	1	a^{15}	0