Lecture 2 Activity Solution

Model 1: Binary Number Representation

1. 2, 3, 3, 4
2. 2, 4, 8
3. 1, 11, 111
4. 16, 32, 64
5. 2X
6. 0, 1, 0, 1, 0, 1, ...
7. 0, 0, 1, 1, 0, 0, 1, 1, ...
8. Every four numbers
9. 1011, 1100, 1101, 1110, 1111
10. 1000001, 1000010, 1000011, 1000100
11. 4, 8, 16, 2^{N-1}
12. 900
13. No
14. Yes, since the weights for the digits in the old number would change.
15. 2^{Index}
16. 25
17. No.
18. Yes, for the same reason as decimal numbers.
19. 37, 59, 65
20. $\sum_{i=0,..m} d_i \times 2^i$
21. $2^2, 2^2 + 2^1, 2^2 + 2^1 + 2^0, 2^4 + 2^2, 2^5 + 2^4$
22. 110000
23. Use a greedy approach. Start with the largest power of 2 possible, and repeat until 0.
24. 11111111, 255.

Group Reflection

1. The range of numbers that can be represented by an N-bit binary number is $[0, 2^N - 1]$.
2. 115
3. 1100011

Model 0x1: Hexadecimal Representation

1. From 0 to 15
2. 16
3. 16^0 (Note that this is always 1, independent of the integer representation.)
4. 16
5. 0x100, 256
6. Hexadecimal
7. 0x15
8. 0x20
9. 48, 90, 285
10. 1100, 1101, 1110, 1111
11. 1100101011111110, 10001011101011011111000000001101
12. Divide the number into blocks with four digits and convert each block according to the table above.
13. 0xD1E, 0x37

Model 1+1: Adding Binary Numbers

14. 81529 + 12256 = 93785 (We are confident that you can do the gradeschool algorithm of adding digit-by-digit)

15. 18 (9+9)

16. The maximum carry value is 1.

17.

b_0	b_1	$b_0 + b_1$
0	0	1
0	1	1
1	0	1
1	1	10

18. The maximum number of bits required to represent the value of adding two 1-bit numbers is 2.

19.

c	b_0	b_1	c + $b_0 + b_1$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	10
1	0	0	1
1	0	1	10
1	1	0	10
1	1	1	11

20. The maximum number of bits required to represent the value of adding two 1-bit numbers and a carry is 2.

21. 1001 + 0101 = 1110

22. $1001_2 = 9_{10}$, $0101_2 = 5_{10}$, $1110_2 = 14_{10}$

23. 1010 + 1100 = 10110

24. 5 bits were required for the result in the previous question.

25. The number of bits needed for the result is at most 1 more than the number of bits in the numbers added together.

26. The maximum number of bits required hold the result of adding two N-bit numbers together is N+1.

27. You might have to truncate the leftmost bit if the result is more than N bits.

Group Reflection

1. 0x6FA = 11011111010_2

2. 1111011_2 = 0x7B

Model 010: Representing Negative Values in Binary

16. The leftmost bit in a non-negative number is 0.

17. 32: 0100000, 42: 0101010

18. 3: 011, -8: 11000

19. If the leftmost bit of a two's complement number is 1, then it's negative. Otherwise, it's non-negative.

20. Step 1: 1111
Step 2: 01111
Step 3: 10000
Step 4: 10001

21.

Decimal	Negative	Invert	Add 1
1	11	00	01
2	110	001	010
3	101	010	011
4	100	011	100
5	1011	0100	0101

22. The "Add 1" column is the same as the "Positive" column in the earlier table.

23. $010111_2 = 23_{10}$, $111010_2 = -6_{10}$

24.

Two's Comp	Decimal
011	3
0011	3
00011	3
101	-3
1101	-3
11101	-3

25. Add one bit to the left that is the same as the sign bit(leftmost bit of original number).

26. Entries 0, 1, 2, 4, and 8 have more bits than required to represent that number. 0: 0, 1: 1, 2: 2, 4: 3, 8: 4.

27.

Bits	Most Positive	Most Negative
1	0	-1
2	1	-2
3	3	-4
4	7	-8

28. The most positive number that can be represented by a N-bit two's complement number is 2^{N-1}-1.

29. The most negative number that can be represented by a N-bit two's complement number is -2^{N-1}.

30. The magnitude of the most positive number is one less than the magnitude of the most negative number for an N-bit two's complement number. This is because we need a way to represent zero.

31. 2^N distinct integers can be represented by an N-bit two's complement representation.

32. $1001 + 0011 = 1100$

33. $1001_2 = -7_{10}$, $0011_2 = 3_{10}$, $1100_2 = -4_{10}$

34. Convert the subtrahend to its negative and add up two numbers.

35. It might be hard to deal with overflow, regular binary addition cannot be used to add two numbers, and we would have two values for zero: "positive" zero and "negative" zero.

Group Reflection

1. $1111000 + 0100111 = 10011111$
$1111000_2 = 120_{10}$, $0100111_2 = 39_{10}$, $10011111_2 = 159_{10}$

2. 1111111111: -1, 0110100010: 418

3. 105: 01101001, -482: 1000011110

4. The range of numbers that can be represented by a 32-bit two's complement number is -2^{31} to $+2^{31}$-1