|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Integer for Binary | Binary | Integer (1’s) | 1’s Complement |  | 2’s Complement |
| 0 | 00000 | 0 | 00000 | 0 | 00000 |
| 1 | 00001 | 1 | 00001 | 1 | 00001 |
| 2 | 00010 | 2 | 00010 | 2 | 00010 |
| 3 | 00011 | 3 | 00011 | 3 | 00011 |
| 4 | 00100 | 4 | 00100 | 4 | 00100 |
| 5 | 00101 | 5 | 00101 | 5 | 00101 |
| 6 | 00110 | 6 | 00110 | 6 | 00110 |
| 7 | 00111 | 7 | 00111 | 7 | 00111 |
| 8 | 01000 | 8 | 01000 | 8 | 01000 |
| 9 | 01001 | 9 | 01001 | 9 | 01001 |
| 10 | 01010 | 10 | 01010 | 10 | 01010 |
| 11 | 01011 | 11 | 01011 | 11 | 01011 |
| 12 | 01100 | 12 | 01100 | 12 | 01100 |
| 13 | 01101 | 13 | 01101 | 13 | 01101 |
| 14 | 01110 | 14 | 01110 | 14 | 01110 |
| 15 | 01111 | 15 | 01111 | 15 | 01111 |
| 16 | 10000 | -15 | 10000 | -16 | 10000 |
| 17 | 10001 | -14 | 10001 | -15 | 10001 |
| 18 | 10010 | -13 | 10010 | -14 | 10010 |
| 19 | 10011 | -12 | 10011 | -13 | 10011 |
| 20 | 10100 | -11 | 10100 | -12 | 10100 |
| 21 | 10101 | -10 | 10101 | -11 | 10101 |
| 22 | 10110 | -9 | 10110 | -10 | 10110 |
| 23 | 10111 | -8 | 10111 | -9 | 10111 |
| 24 | 11000 | -7 | 11000 | -8 | 11000 |
| 25 | 11001 | -6 | 11001 | -7 | 11001 |
| 26 | 11010 | -5 | 11010 | -6 | 11010 |
| 27 | 11011 | -4 | 11011 | -5 | 11011 |
| 28 | 11100 | -3 | 11100 | -4 | 11100 |
| 29 | 11101 | -2 | 11101 | -3 | 11101 |
| 30 | 11110 | -1 | 11110 | -2 | 11110 |
| 31 | 11111 | -0 | 11111 | -1 | 11111 |

1. Binary, 1’s Complement, & 2’s Complement Table
2. The first electronic word processor, the Wang 1200, could handle one line of text at a time, which could be edited before printing or storing on as data on a cassette tape. Each line of text was stored in 100 Bytes, and could be a maximum of 100 characters long.
   1. How many bits per character is that?
      1. 1 byte = 8 bits
      2. 1 byte = 1 character
         1. This is because 100 bytes = 100 characters.
         2. Bytes/character = 100 bytes/100 characters = 1 byte/character
      3. Answer: 1 byte/character = 8 bits/character.
   2. How many possible symbols could be represented in this way?
      1. We know that we have 8 bits. The maximum we can store is 11111111 in binary and 00000000 in binary. Numerically, the max number is 255 and 0 is the lowest. Because of this, we can represent 256 characters with 8 bits in binary.
   3. How many bits would it take to represent all of the lowercase letters in the Latin alphabet used for English?
      1. There are 26 letters in the alphabet. Simply, our max number needs to be 25 because we can store a character starting at 0.
      2. The binary for 25 is 11001, which is 5 bits.
      3. Answer: 5 bits.
   4. What about upper and lower case?
      1. Upper case alphabet + lower case alphabet = total characters
      2. 26 + 26 = 52
      3. We need a max value of 51 and store first character as 0.
      4. The binary for 51 is 110011. This is 5 bits.
      5. Answer: 6 bits.
   5. What about all of the letters, numbers, and punctuation on your computer’s keyboard?
      1. 10 Number keys (0-9) = 10 keys
      2. Punctuation keys = 21 keys
      3. Space, enter, tab = 1 key
      4. Upper and lowercase letters = 52 keys
      5. Total = 84 keys
      6. We need a binary number going up to 83.
      7. Binary for 83 = 1010011
      8. Answer: 7 bits
3. Voided by TA
4. Binary Arithmetic
   1. 5 + 5 = 10

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0 | 1 | 0 |
| 23 | 22 | 21 | 20 |

* + 1. 23 + 21 = 10
    2. Binary = 1010
  1. 8 + 7 = 15

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |
| 23 | 22 | 21 | 20 |

* + 1. 23 + 22 + 21 + 20 = 15
    2. Binary = 1111
  1. 9 – 3 = 6

|  |  |  |
| --- | --- | --- |
| 1 | 1 | 0 |
| 22 | 21 | 20 |

* + 1. 22 + 21 + 20 = 6
    2. Binary = 110
  1. 7 – 5 = 2

|  |  |
| --- | --- |
| 1 | 0 |
| 21 | 20 |

* + 1. 21 + 20 = 2
    2. Binary = 10

1. Converting back to decimal form
   1. 1011

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0 | 1 | 1 |
| 23 | 22 | 21 | 20 |

* + 1. 23 + 21 + 20 = 8 + 2 + 1 = 11
  1. 1010000

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 26 | 25 | 24 | 23 | 22 | 21 | 20 |

* + 1. 26 + 24 = 64 + 16 = 80
  1. 11001111

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |

* + 1. 27 + 26 + 23 + 22 + 21 + 20 = 128 + 64 + 8 + 4 + 2 = 206
  1. 1010000011101000

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 215 | 214 | 213 | 212 | 211 | 210 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |

* + 1. 215 + 213 + 27 + 26 + 25 + 23 = 32,768 + 8,192 + 128 + 64 + 32 + 8 = 41,192