CS Basics - Exercises
Play with different bases

E. Benoist

Fall Term 2022-23

1 Hexadecimal to decimal

Exercise 1. Compute the decimal values for the following Hexadecimal numbers.

- `0x99`
- `0xA5C`
- `0x700`
- `0xE03`
- `0xBBF9`
- `0xC9`

Exercise 2.

- `0xFFFF`
- `0x1000`
- `0x1119`
- `0x2345`
- `0x739AE`

Exercise 3.

- `0xFF`
- `0xCA45C`
- `0xC8900`
- `0xCC`
- `0x8B359002`

2 Decimal to Hexadecimal

Exercise 4. Write the following decimal numbers into hexadecimal

- 304
- 4095
- 31'155
- 1'011'001
- 12'345'678

3 Binary

Exercise 5. Convert into decimal the following binary values

- 0b101
- 0b1100
- 0b1000111
- 0b101010101
- 0b11011011
- 0b10001

Exercise 6. Convert into binary the following numbers

- 5
- 10
- 25
- 1000
- 245

4 Hexadecimal to binary (and vice versa)

Exercise 7. Write in binary the following hex numbers

- `0x0A`
- `0x10`
- `0x04`
- `0xAA`
- `0xA0`
- `0x40`
- `0x5 73 CD`
- `0xC FF 32`
- `0x4 0A A0 01 10 0A`
- `0x40 AA 00 11 00 A4 0A A0 01 10 0A`

Exercise 8. Write in hex the following binary numbers

- `0b1000`
- `0b1111`
- `0b1001`
- `0b0001`
- `0b0011`
- `0b1010 0011`
- `0b0111 0111`
- `0b1101 0111`

5 Octal to decimal

Exercise 9. Convert the following octal numbers into decimal

- 100
- 6573
- 7777

- 12345
- 100000

Exercise 10. Write the following decimal numbers into octal

- 20
- 350
- 1000
- 890

6 Operations using Hex numbers

Exercise 11. Add the following numbers:

- `0xA23 + 0x5BB`
- `0xDE56F3 + 0x78FFE`
- `0xAAAAAAA + 0x1234567`
- `0xFEA + 0x123A`
- `0xA00A + 0xFFFA`

Exercise 12. Perform the following subtractions (all numbers are given in hexadecimal)

- `0xABCDEFAA - 0x12345678`
- `0xBAD343 - 0xBACCAA`
- `0x123456 - 0xAAAAA`
- `0x12F36AE - 0x0AAAAF`

7 Additions

Perform the following additions in hexadecimal

1. 0xA67B+0xFFFFF
2. 0xAFFF+0x1001
3. 0xBF12+0x12BF
4. 0x1234+0xFEDC
5. 0xFFEE+0xFFEE
6. 0xFFEE+0xFFEE+0xFFEE

8 Logical operations

In programming, we will extend the logical operations AND, OR, XOR to arrays of bits.
For instance the AND operator :

```
    0b 1011 0101
AND 0b 1110 1110
    ---------
    0b 1010 0100
```

Exercise 13. Compute the following results using the logical operators on arrays of bits

1. 0b 1001 1110 AND 0b 0011 1001 =
2. 0b 0111 1101 AND 0b 1111 0000 =
3. 0b 1100 1001 AND 0b 1111 0010 =
4. 0b 1111 1001 AND 0b 1011 0100 =
5. 0b 0000 1000 AND 0b 1101 1000 =
6. 0b 1001 1110 OR 0b 0011 1001 =
7. 0b 0111 1101 OR 0b 1111 0000 =
8. 0b 1100 1001 OR 0b 1111 0010 =
9. 0b 1111 1001 OR 0b 1011 0100 =
10. 0b 0000 1000 OR 0b 1101 1000 =
11. 0b 1001 1110 XOR 0b 0011 1001 =
12. 0b 0111 1101 XOR 0b 1111 0000 =
13. 0b 1100 1001 XOR 0b 1111 0010 =
14. 0b 1111 1001 XOR 0b 1011 0100 =
15. 0b 0000 1000 XOR 0b 1101 1000 =
16. NOT 0b 0000 1010 =
17. NOT 0b 1010 1110 =
18. NOT 0b 0001 1110 =
19. NOT 0b 1111 0000 =

Exercise 14. The following exercise is the same as the previous one, using binary operators on hexadecimal numbers. They must be interpreted as arrays of bits.

1. 0xA1 AND 0b00111001 =
2. 0xAA AND 0b11110000 =
3. 0xAC AND 0xFF =
4. 0x10 AND 0x35 =
5. 0x5C AND 0x3F =

6. 0xEE OR 0b00111001 =
7. 0xE0 OR 0b11110000 =
8. 0xFC OR 0x00 =
9. 0xCF OR 0xD0 =
10. 0x35 OR 0x57 =
11. 0xDF XOR 0b00111001 =
12. 0xD1 XOR 0b11110000 =
13. 0xED XOR 0x00 =
14. 0xB0 XOR 0xD0 =
15. 0x26 XOR 0x57 =
16. NOT 0x29 =
17. NOT 0x09 =
18. NOT 0xFF =
19. NOT 0x01 =