1. Convert (D1AF)16 to binary and decimal. Convert (11011101001)2 to decimal and hexadecimal. [10]

2. Using 4-bit Twos complement, write the binary number for +5 and -4. [5]

3. Write Python or Java code to convert a binary number to decimal and vice versa. [10]

4. Write Python or Java code to convert hexa-decimal number to binary and vice versa. [10]

5. Draw the memory layout for the byte-sized (all elements are 1 byte wide) list, pixel = { 0x10, 0xFF, 0x56}. The base address is 0x1FFF. [10]

6. Draw the memory layout for the half-word-sized (all elements are 2 byte wide) list, pixel = { 0x1000, 0x03, 0x56FF}. The base address is 0x1FF0. [10]

Denary	Binary	Hex
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

1) Convert (D1AF)16 to binary and decimal. Convert (11011101001)2 to decimal and hexadecimal.
   - Binary: D = 1101. 1 = 1. A = 1010. F = 1111. 1 becomes 0001 to make it 4 long. 1101000110101111
   - Decimal: D = 13. 1 = 1. A = 10. F = 15. 13 times 16^3 + 1 times 16^2 + 10 times 16^1 + 15 times 16^0 = 53,679 to the base of 10
   - (11011101001)2 to decimal. 1 times 2^0 + 0 times 2^1 + 0 times 2^2 + 1 times 2^3 + 0 times 2^4 + 1 times 2^5 + 1 times 2^6 + 1 times 2^7 + 0 times 2^8 + 1 times 2^9 + 1 times 2^10 = 1769 to the base of 10.
   - (11011101001)2 to hexadecimal. 110 = 0110 = 6. 1110 = E. 1001 = 9. 1001 = 9. 6E9 to the base of 16.
2) Using 4-bit Twos complement, write the binary number for +5 and -4
   - +5 = -5 = 1011. Invert = 0100 + 1 = 0101.
   - -4 = 4 = 0100. Invert is 1011 and add 1 to equal 1100.
3) Python Code

4) Python Code
5) Image below
6) Image below