Tran Thanh Duong (SE160185) - AI1602

Assignment (1 & 2)

1. Convert the following binary numbers to decimal without using a calculator, showing your work (2 marks):

a. $(0110101)_2 = (0 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = (53)_{10}$

b. $(10110001)_2 = (1 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = (177)_{10}$

c. $(01111110.01)_2 = (0 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) + (0 \times 2^{-1}) + (1 \times 2^{-2}) = (126.25)_{10}$

d. $(111111.011)_2 = (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (0 \times 2^{-1}) + (1 \times 2^{-2}) + (1 \times 2^{-3}) = (63.375)_{10}$

2. Convert the following decimal numbers to binary without using a calculator, showing your work (2 marks):

a. 1234.75

First, we multiply the 1234.75 by 2^2 to become an integer number: $1234.75 \times 2^2 =$ **4939**

Then, divide **4939** by the base 2 to get remainders as the digits:

Bit	Divided by 2	Quotient	The Remainder
0	4939/2	2469	**1**
1	2469/2	1234	**1**
2	1234/2	617	**0**
3	617/2	308	**1**
4	308/2	154	**0**
5	154/2	77	**0**
6	77/2	38	**1**
7	38/2	19	**0**
8	19/2	9	**1**
9	9/2	4	**1**
10	4/2	2	**0**
11	2/2	1	**0**
12	1/2	0	**1**

=> $(4939)_{10} = (1001101001011)_2$

Finally, shift 2 bits to the left to get the final result:

$(1234.75)_{10} = (10011010010.11)_2$

b. 88.250

First, we multiply the 88.250 by 2^2 to become an integer number: $88.250 \times 2^2 =$ **353**

Then, divide **353** by the base 2 to get remainders as the digits:

Bit	Divided by 2	Quotient	The Remainder
0	353/2	176	**1**
1	176/2	88	**0**
2	88/2	44	**0**
3	44/2	22	**0**
4	22/2	11	**0**
5	11/2	5	**1**
6	5/2	2	**1**
7	2/2	1	**0**
8	1/2	0	**1**

=> $(353)_{10} = (101100001)_2$
Finally, shift 2 bits to the left to get the final result:
$(88.250)_{10} = (1011000.01)_2$

c. 124.02
First, we multiply the 124.02 by 2^8 to become an (approximate) integer number:
$124.02 \times 2^8 \approx$ **31749**
Then, divide **31749** by the base 2 to get remainders as the digits:

Bit	Divided by 2	Quotient	The Remainder
0	31749/2	15874	**1**
1	15874/2	7937	**0**
2	7937/2	3968	**1**
3	3968/2	1984	**0**
4	1984/2	992	**0**
5	992/2	496	**0**
6	496/2	248	**0**
7	248/2	124	**0**
8	124/2	62	**0**
9	62/2	31	**0**
10	31/2	15	**1**
11	15/2	7	**1**
12	7/2	3	**1**
13	3/2	1	**1**

Bit	Divided by 2	Quotient	The Remainder
14	1/2	0	**1**

=> $(31749)_{10} = (111110000000101)_2$
Finally, shift 8 bits to the left to get the final result:
$(124.02)_{10} \approx (1111100.00000101)_2$

d. 140.075
First, we multiply the 140.075 by 2^8 to become an (approximate) integer number:
$140.075 \times 2^8 \approx$ **35859**
Then, divide **35859** by the base 2 to get remainders as the digits:

Bit	Divided by 2	Quotient	The Remainder
0	35859/2	17929	**1**
1	17929/2	8964	**1**
2	8964/2	4482	**0**
3	4482/2	2241	**0**
4	2241/2	1120	**1**
5	1120/2	560	**0**
6	560/2	280	**0**
7	280/2	140	**0**
8	140/2	70	**0**
9	70/2	35	**0**
10	35/2	17	**1**
11	17/2	8	**1**
12	8/2	4	**0**
13	4/2	2	**0**
14	2/2	1	**0**
15	1/2	0	**1**

=> $(35859)_{10} = (1000110000010011)_2$
Finally, shift 8 bits to the left to get the final result:
$(140.075)_{10} \approx (10001100.00010011)_2$

3. The following are sign-and-magnitude binary numbers in 8-bit location. Convert them to decimal without using a calculator, showing your work (3 marks):
a. 01110111
• The leftmost bit of this number is **0**, it means this number is **positive**.

• Convert 7-bit binary to decimal:
$(1110111)_2 = (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) = (119)_{10}$
Therefore, **01110111** in sign-and-magnitude binary number is equal to **+119** in decimal.

b. 11111100
• The leftmost bit of this number is **1**, it means this number is **negative**.
• Convert 7-bit binary to decimal:
$(1111100)_2 = (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (0 \times 2^0) = (124)_{10}$
Therefore, **11111100** in sign-and-magnitude binary number is equal to **-124** in decimal.

c. 01110100
• The leftmost bit of this number is **0**, it means this number is **positive**.
• Convert 7-bit binary to decimal:
$(1110100)_2 = (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (0 \times 2^0) = (116)_{10}$
Therefore, **01110111** in sign-and-magnitude binary number is equal to +**116** in decimal.

4. Convert the following decimal integers to sign-and-magnitude with 8-bit allocation without using a calculator, showing your work (3 marks):
a. 154
154 is **out of range** (-128 to 127) so it cannot be converted to sign-and-magnitude with 8-bit allocation.
b. -107
• Convert 107 to 7-bit binary: $(107)_{10} = (1101011)_2$
• Add sign to number: 11101011
Therefore, **-107** in decimal is equal **11101011** in sign-and-magnitude binary number.

c. -5
• Convert 5 to 7-bit binary: $(7)_{10} = (0000101)_2$
• Add sign to number: 10000101
Therefore, **-5** in decimal is equal **10000101** in sign-and-magnitude binary number.

5. Do the following conversions
a. (e1) : 1234d = 10011010010b = 4D2h
b. (e2) : 2567d = 101000000111b = A07h
c. (e3) : 101011001110b = ACEh = 2766d
d. (e4) : 1100110011101101b = CCEDh = 52461d
e. (e5) : 17ADh = 1011110101101b = 6061d
f. (e6) : 5DE6h = 101110111100110b = 24038d
g. (e7) : 9ABF2h = 10011010101111110010b = 633842d
h. (e8) : 101011001110b = ACEh = 2766d
i. (e9) : 110010011101b = C9Dh = 3229d
j. (e10) : 1001011011011b = 12DBh = 4827d

6. Describe the steps for performing the two algorithms Selection Sort and Bubble Sort for the following sequence of numbers: 5, 8, 9, 3, 7, 2, 6, 12, 18, 4

In **Selection Sort**, the list is divided into 2 parts: sorted and unsorted. After every pass, the smallest element from the unsorted list will be added to sorted list and that element will be removed from the unsorted list. The algorithm of **Selection Sort** is described in 2 steps repeatedly until the list is sorted:

- Step 1: Find the smallest element in the unsorted list.
- Step 2: Swap that element with the first element in the unsorted list, the first index of unsorted list is shifted one position to the right.

Performing **Selection Sort**:

5	8	9	3	7	2	6	12	18	4
2	8	9	3	7	5	6	12	18	4
2	8	9	3	7	5	6	12	18	4
2	3	9	8	7	5	6	12	18	4
2	3	9	8	7	5	6	12	18	4
2	3	4	8	7	5	6	12	18	9
2	3	4	8	7	5	6	12	18	9
2	3	4	5	7	8	6	12	18	9
2	3	4	5	7	8	6	12	18	9
2	3	4	5	6	8	7	12	18	9
2	3	4	5	6	7	8	12	18	9
2	3	4	5	6	7	8	12	18	9
2	3	4	5	6	7	8	12	18	9
2	3	4	5	6	7	8	12	18	9
2	3	4	5	6	7	8	12	18	9
2	3	4	5	6	7	8	9	18	12
2	3	4	5	6	7	8	9	18	12

Tran Thanh Duong (SE160185) - AI1602

2	3	4	5	6	7	8	9	12	18
2	3	4	5	6	7	8	9	12	18
2	3	4	5	6	7	8	9	12	18

In every pass in **Bubble Sort**, the adjacent elements are swapped if they are unsorted and finally larger one being shifted to the end of the list.

Performing **Bubble Sort**:

5	8	9	3	7	2	6	12	18	4
5	8	3	7	2	6	9	12	4	18
5	3	7	2	6	8	9	4	12	18
3	5	2	6	7	8	4	9	12	18
3	2	5	6	7	4	8	9	12	18
2	3	5	6	4	7	8	9	12	18
2	3	5	4	6	7	8	9	12	18
2	3	4	5	6	7	8	9	12	18
2	3	4	5	6	7	8	9	12	18
2	3	4	5	6	7	8	9	12	18