Jose Franco
Baquera
1/18/18
CS 273

Homework #1

① Some computer scientist claims he is 1,000,000 years old. The catch is that the age is in binary. How old is this person in decimal years?

Base:	2	2	2	2	2	2	2
Age in Binary:	1	0	0	0	0	0	0
Slots:	6	5	4	3	2	1	0

$Sum = 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0$
$= 64 + 0 + 0 + 0 + 0 + 0 + 0 = 64$

Answer: 64
The computer scientist is 64 in decimal years.

② Convert the following three numbers into octal numbers. You must show the procedure to obtain full credits: 0xFC1, 911, 0b0011010101010

- 0xFC1

First convert 0xFC1 into decimal.

Base:	16	16	16
Hex:	F	C	1
Slots:	2	0	1

$Sum = 15 \cdot 16^2 + 12 \cdot 16^1 + 1 \cdot 16^0 = 3840 + 192 + 1 = 4033_{10}$

Now convert 4033_{10} into octal by repeated division.

4033/8 = 504 R1 (Least Significant)
504/8 = 63 R0
63/8 = 7 R7
7/8 = 0 R7 (Most Significant)

Must be read "bottom to top" 7701

Answer: 0xFC1 = 07701

- 911

Convert it into octal by repeated division.

911/8 = 113 R7 (Least Significant)
113/8 = 14 R1
14/8 = 1 R6
1/8 = 0 R1 (Most Significant)

Must be read "bottom to top"
1617

$\boxed{\text{Answer: } 911 = 01617}$

- 0b0011010101010

First convert the binary number into a decimal

Base : 2 2 2 2 2 2 2 2 2 2 2
Binary : 0 0 1 1 0 1 0 1 0 1 0
Slots : 10 9 8 7 6 5 4 3 2 1 0

$\text{Sum} = 1 \cdot 2^8 + 1 \cdot 2^7 + 1 \cdot 2^5 + 1 \cdot 2^3 + 1 \cdot 2^1$
$= 256 + 128 + 32 + 8 + 2 = 426$

Now convert 426 into octal by repeated division

426/8 = 53 R2 (Least Significant)
53/8 = 6 R5
6/8 = 0 R6 (Most Significant)

Must be read "bottom to top"
652

$\boxed{\text{Answer: } 0b0011010101010 = 0652}$

③ Convert the following binary <u>unsigned</u> numbers to octal, hexadecimal, and decimal:

1101, 0100, 1011

101, 0001

10110

- 1101, 0100, 1011

- Octal: First convert the number into decimal

Base:	2	2	2	2	2	2	2	2	2	2	2	2
Binary:	1	1	0	1	0	1	0	0	1	0	1	1
Slots:	11	10	9	8	7	6	5	4	3	2	1	0

Sum $= 1\cdot 2^{11} + 1\cdot 2^{10} + 1\cdot 2^{8} + 1\cdot 2^{6} + 1\cdot 2^{3} + 1\cdot 2^{1} + 1\cdot 2^{0} = 3403$

Now convert 3403 into octal by repeated division

3403 / 8 = 425 R3 (<u>Least significant</u>)

425 / 8 = 53 R1

53 / 8 = 6 R5

6 / 8 = 0 R6 (<u>Most significant</u>)

Must be read "bottom to top" 6513

$\boxed{\text{Answer: } 0b110101001011 = 06513}$

- Hex: Divide the binary number into nibbles

1101	0100	1011	
D	4	B	⇒ D4B

$\boxed{\text{Answer: } 0b110101001011 = 0xD4B}$

- Decimal: Use counting by weights

Base:	2	2	2	2	2	2	2	2	2	2	2	2
Binary:	1	1	0	1	0	1	0	0	1	0	1	1
Slots:	11	10	9	8	7	6	5	4	3	2	1	0

Sum $= 1\cdot 2^{11} + 1\cdot 2^{10} + 1\cdot 2^{8} + 1\cdot 2^{6} + 1\cdot 2^{3} + 1\cdot 2^{1} + 1\cdot 2^{0}$

$= 2048 + 1024 + 256 + 64 + 8 + 2 + 1 = 3403$

$\boxed{\text{Answer: } 0b110101001011 = 3403}$

- 101, 0001

- Octal: First convert the number into decimal

Base: 2 2 2 2 2 2 2
Binary: 1 0 1 0 0 0 1
Slots: 6 5 4 3 2 1 0

Sum $= 1 \cdot 2^6 + 1 \cdot 2^4 + 1 \cdot 2^0 = 64 + 16 + 1 = 81$

Now convert 81 into octal by repeated division

81/8 = 10 R1 (Least Significant)
10/8 = 1 R2
1/8 = 0 R1 (Most Significant)

Must be read "bottom to top"
121

$\boxed{\text{Answer: } 0b1010001 = 0121}$

- Hex: Divide the binary number into nibbles.

0101 | 0001
5 | 1 ⇒ 51

$\boxed{\text{Answer: } 0b1010001 = 0x51}$

- Decimal: Using counting by weights

Base: 2 2 2 2 2 2 2
Binary: 1 0 1 0 0 0 1
Slots: 6 5 4 3 2 1 0

Sum $= 1 \cdot 2^6 + 1 \cdot 2^4 + 1 \cdot 2^0 = 64 + 16 + 1 = 81$

$\boxed{\text{Answer: } 0b1010001 = 81}$

- 1, 0110

- Octal: First convert the number into decimal.

Base: 2 2 2 2 2
Binary: 1 0 1 1 0
Slots: 4 3 2 1 0

Sum $= 1\cdot 2^4 + 1\cdot 2^2 + 1\cdot 2^1 = 16 + 4 + 2 = 22$

Now convert 22 to octal by repeated division

$22/8 = 2$ R 6 (Least Significant)
$2/8 = 0$ R 2 (Most Significant)

Answer: 0b10110 = 026

- Hex: Divide the binary number into nibbles

0001 | 0110
1 | 6 ⇒ 16

Answer: 0b10110 = 0x16

- Decimal: Using counting by weights

Base: 2 2 2 2 2
Binary: 1 0 1 1 0
Slots: 4 3 2 1 0

Sum $= 1\cdot 2^4 + 1\cdot 2^2 + 1\cdot 2^1 = 16 + 4 + 2 = 22$

Answer: 0b10110 = 22

④ Convert the following Signed two's binary complement binary numbers to decimal.

1011, 0100, 1011

101, 0001

10000

11111

- 101101001011

MSB = 1

$-X = 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1$

One's Complement

$-X = 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0$

Two's Complement $+\ 1$

$-X = 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1$

$\cancel{2048}$ 1024 $\cancel{512}$ $\cancel{256}$ 128 $\cancel{64}$ 32 16 $\cancel{8}$ 4 $\cancel{2}$ 1

Sum = 1024 + 128 + 32 + 16 + 4 + 1 = 1205

$-X = 1205_{10} \Rightarrow X = -1205_{10}$

$\boxed{\text{Answer: } 101101001011_2 = -1205_{10}}$

- 101, 0001 MSB = 1

$-X = 1010001$

$-X = 0101110$ (one's complement)

$+\ 1$

0101111 (Two's Complement)

$\cancel{64}$ 32 $\cancel{16}$ 8 4 2 1

Sum = 32 + 8 + 4 + 2 + 1 = 47

$-X = 47_{10} \Rightarrow X = -47_{10}$

$\boxed{\text{Answer: } 1010001_2 = -47_{10}}$

- 1 0 0 0 0

  MSB = 1

  $-x = 1\ 0\ 0\ 0\ 0$

  $-x = 0\ 1\ 1\ 1\ 1$ (one's complement)

  $+\quad\quad\quad\quad 1$ (Two's complement)

  $1\ 0\ 0\ 0\ 0$

  $16\ \cancel{8}\ \cancel{4}\ \cancel{2}\ \cancel{1}$

  Sum = 16

  $-x = 16_{10} \Rightarrow x = -16_{10}$

  $\boxed{\text{Answer: } 10000_2 = -16_{10}}$

- 1 1 1 1 1

  MSB = 1

  $-x = 1\ 1\ 1\ 1\ 1$

  $-x = 0\ 0\ 0\ 0\ 0$ (one's complement)

  $+\quad\quad\quad\quad 1$

  $0\ 0\ 0\ 0\ 1$ (two's complement)

  $\cancel{16}\ \cancel{8}\ \cancel{4}\ \cancel{2}\ 1$

  Sum = 1

  $-x = 1_{10} \Rightarrow x = -1_{10}$

  $\boxed{\text{Answer: } 11111_2 = -1_{10}}$

⑤ Why do we use 2's complement instead of 1's complement numbers to represent negative numbers?

There are many reasons why we use 2's complement instead of 1's complement numbers to represent negative numbers. For starters, the 2's complement has only <u>one</u> value for zero while 1's complement has <u>two</u> values for zero (i.e. negative zero - 11111111 and positive zero - 00000000). Therefore, 2's complement makes addition and subtraction easier since there is no need for an "end-around" carry value. Another reason why we use 2's complement to represent negative numbers it that the summation of a binary number and its corresponding 2's complement is always zero, which makes it easier for a computer to do arithmetic. It is also important to note that arithmetic by 2's complement can be done completely ignoring whether a number is signed or unsigned. This means that the CPU does not care if you are using signed numbers or unsigned numbers. In essence, we use 2's complement instead of 1's complement to represent negative numbers because it makes it easier for the CPU to perform arithmetic calculations like addition and subtraction.

(6) Find a computer being developed that is entirely different from the modern electronic computer. Describe using your own language how it works and provide references the the new computer. Please be careful in choosing a trustworthy source.

One type of computer being developed that is entirely different from the modern electronic computer is the Quantum Computer. Google, IBM, and Intel are just a few of the many tech companies that are researching and engineering these computers. In essence, Quantum Computers use a computing unit called a "quibit". Modern electronic computers don't use "quibits" but rather semiconductors, which are only capable of representing information as a series of 1's and 0's. A "quibit", however, can represent both a 1 and 0 simultaneously! In other words, two "quibits" can represent 10, 11, 01, and 00 at the same time. Quantum Computers are of a great interest because they are capable of solving the hardest mathematical and computing problems found at present time. These problems are too complex and difficult to be solved by the modern electronic computer. It is important to note that because "quibits" exhibit quantum properties, Quantum Computers become extremly hot, so hot that they need to run inside a freezer. Another downside of Quantum Computers is that they can only maintain a quantum computing state for a few microseconds at a time. Quantum Computers can also be used to easily crack most forms of encryption used today, which poses a huge national security risk for countries like the United States.

Reference:

Wadhwa, V. (2018). Commentary: These Next-Generation Supercomputers Are So Hot They Need to Run in a Freezer. Fortune. Retrived January 27, 2018, from www.fortune.com/2018/01/17/what-is-quantum-computing/