Solutions to Week 2 Introduction and Number Systems

Question 1

What is the largest number that you can get with 4 bits, with 8 bits and 16 bits?

Answer: The maximum/largest **unsigned** number stored in an n-bit word is 2^n-1.

1. A 4-bit word stores 16 numbers that between 0 and 15 (2^4-1) **inclusive.**
2. A 8-bits word stores 256 numbers that between 0 and 255 (2^8 - 1) **inclusive.**
3. A 16-bit word stores 65536 numbers that between 0 and 65535 (2^{16} - 1) **inclusive.**

Question 2 — Number System Conversion

Decimal to Other Base System (binary, octal, and hexadecimal)

Convert the following decimal numbers to binary, octal, and hexadecimal:

1. 117_{10}

Answer:

Decimal to binary (base 2) conversion

	Quotient	Remainder
117/2	58	1
58/2	29	0
29/2	14	1
14/2	7	0
7/2	3	1
3/2	1	1
1/2	**0**	1

You stop when 0 is obtained.

Read up the Remainder column 117_{10} =$111\ 0101_2$ = 165_8 = 75_{16}

Binary to octal/hex conversion: The binary number is cut into groups of three/four, starting from the **right.** Add extra zeroes to the front of the first number to fill out the last group of three/four if necessary. Then replace each 3-digit/4-digit group with the equivalent octal/hex digit.

2. 127_{10}

Answer: 127_{10} = $0111\ 1111_2$ = 177_8 = $7F_{16}$

3. 128_{10}

Answer: 128_{10} =$1000\ 0000_2$ = 200_8 = 80_{16}

4. 255_{10}

Answer: 255_{10} =$1111\ 1111_2$ = 377_8 = FF_{16}

Other Base System (binary, octal, and hexadecimal) to Decimal

1. Convert 1101_2 to decimal

   Answer: $1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1 = 13_{10}$

2. Convert 7014_8 to decimal

   Answer: $7014_8 = 7 \times 8^3 + 0 \times 8^2 + 1 \times 8^1 + 4 \times 8^0 = 3596_{10}$

3. Convert $7DE_{16}$ to decimal

   Answer: $7DE_{16} = 7 \times 16^2 + 13 \times 16^1 + 14 \times 16^0 = 1792 + 208 + 14 = 2014_{10}$

Other Base System (decimal, binary, octal, and hexadecimal) to Non-Decimal

1. Convert 217_{10} to base 7

   Answer:

	Quotient	Remainder
217/7	31	0
31/7	4	3
4/7	**0**	4

   You stop when 0 is obtained.

   Read up the Remainder column $217_{10} = 430_7$

2. Convert 1101_2 to base 5

   Answer:

   Step 1: convert 1101_2 to decimal: $1101_2 = 13_{10}$

   Step 2: convert 13_{10} to base 5: $13_{10} = 23_5$

3. Convert 7014_8 to base 9

   Answer:

   Step 1: convert 7014_8 to decimal: $7014_8 = 3596_{10}$

   Step 2: convert 3596_{10} to base 9: $3596_{10} = 4835_9$

4. Convert $7DE_{16}$ to base 6

   Answer:

   Step 1: convert $7DE_{16}$ to decimal: $7DE_{16} = 2014_{10}$

   Step 2: convert 2014_{10} to base 6: $2014_{10} = 13154_6$

Binary <-> Octal, Binary <-> Hexadecimal

1. Convert $11100\ 1010\ 1110\ 1111\ 1111_2$ to octal

   Answer:

   The binary number is cut into groups of **three, starting from the right**. Add extra zeroes to the front of the first number to fill out the last group of three if necessary. Then replace each **3-digit** group with the equivalent **octal** digit.

   $11100\ 1010\ 1110\ 1111\ 1111_2 = 111\ 001\ 010\ 111\ 011\ 111\ 111_2 = 7127377_8$

2. Convert $1010\ 1001\ 0101\ 1111\ 1000_2$ to hexadecimal

    Answer:

    The binary number is cut into groups of **four**, **starting from the right**. Add extra zeroes to the front of the first number to fill out the last group of three if necessary. Then replace each **4-digit** group with the equivalent **hexadecimal** digit.

    $1010\ 1001\ 0101\ 1111\ 1000_2 = A95F8_{16}$

3. Convert 671_8 to binary

    Answer:

    Convert each octal digit to a **3-digit** binary number. Combine all the resulting binary groups (of 3 **digits** each) into a single binary number.

    $671_8 = 110\ 111\ 001_2$

4. Convert $DEADFACE_{16}$ to binary

    Answer:

    Convert each hexadecimal digit to a **4-digit** binary number. Combine all the resulting binary groups (of 4 **digits** each) into a single binary number.

    $DEADFACE_{16} = 1101\ 1110\ 1010\ 1101\ 1111\ 1010\ 1100\ 1110_2$

Question 3 — Binary Addition

Add the following 8 bit numbers and state whether the answer is valid to 8-bit arithmetic.

Show your working especially any "carries".

1. $1111\ 0000_2 + 1111\ 1111_2$

    Answer:

Carry	1	1	1	1	0	0	0	0	
		1	1	1	1	0	0	0	0
		1	1	1	1	1	1	1	1
Result	1	1	1	1	0	1	1	1	1

An overflow has occurred, the carry is in 9th column, the result will not fit into 8 bits, so it is invalid.

2. $0111\ 1111_2 + 0011\ 1111_2$

    Answer:

Carry		1	1	1	1	1	1	1	
		0	1	1	1	1	1	1	1
		0	0	1	1	1	1	1	1
Result		1	0	1	1	1	1	1	0

No overflow has occurred, the result fits into 8 bits, so the result is valid.

3. $0111\ 0000_2 + 1111\ 0000_2$

Answer:

Carry	1	1	1	1	0	0	0	0	
		0	1	1	1	0	0	0	0
		1	1	1	1	0	0	0	0
Result	1	0	1	1	0	0	0	0	0

An overflow has occurred, the carry is in 9th column, the result will not fit into 8 bits, so it is invalid.

Question 4 — Binary Negative Numbers

1. To get the two's complement negative notation of an integer, you write out the number in binary. You then invert the digits, and add one to the result.

   Show how -27_{10} would be expressed in two's complement notation.

   Answer:

   1. Convert decimal to binary: 27_{10} in binary is: $0001\ 1011_2$
   2. Invert the digits: $1110\ 0100_2$
   3. Add one — $1110\ 0101_2$

   $-27_{10} = 1110\ 0101_2$

2. Our numbers are 8-bits long, suppose we want to subtract 27_{10} from 115_{10}, show how to perform binary subtraction using the two's complement method.

   Answer:

   $115_{10} - 27_{10} = 115_{10} + (-27_{10})$

   $115_{10} = 0111\ 0011_2$

   $-27_{10} = 1110\ 0101_2$

Carry	1	1	1	0	0	1	1	1	
		0	1	1	1	0	0	1	1
		1	1	1	0	0	1	0	1
Result	1	0	1	0	1	1	0	0	0

The 9th overflow bit is disregarded as we are only interested in the first 8-bits, so the result is: $0101\ 1000_2$ or $(64 + 16 + 8) = 88_{10}$

Question 5 — Hexadecimal Addition

Add the following 8-bit numbers and state whether the answer is valid to 16-bit arithmetic.

Show your working especially any "carries".

1. $1ABC_{16} + 1234_{16}$

Answer:

Method 1

Step 1: convert each hexadecimal digit to binary:

$1ABC_{16} = 0001\ 1010\ 1011\ 1100_2$

$1234_{16} = 0001\ 0010\ 0011\ 0100_2$

Step 2: apply binary addition

Carry	0	0	1	0	0	1	0	0	0	1	1	1	1	0	0	
	0	0	0	1	1	0	1	0	1	0	1	1	1	1	0	0
	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
Result	0	0	1	0	1	1	0	0	1	1	1	1	0	0	0	0

Step 3: convert the binary result to hexadecimal: $0010\ 1100\ 1111\ 0000_2 = 2CF0_{16}$

Method 2

The students **do not** need to know how to add 2 hexadecimal directly but here is the solution.

Carry	0	0	1	
	1	A	B	C
	1	2	3	4
Result	2	C	F	0

(The students will struggle with C+4 but change to decimal in your head $12_{10} + 4_{10} = 16_{10} = 10_{16}$.)

The result fits into 16 bits.

2. $ABBA_{16} + CAFE_{16}$

Answer:

Method 1

Step 1: convert each hexadecimal digit to binary:

$ABBA_{16}$ = $1010\ 1011\ 1011\ 1010_2$

$CAFE_{16}$ = $1100\ 1010\ 1111\ 1110_2$

Step 2: apply binary addition

Carry	1	0	0	0	1	0	1	1	1	1	1	1	1	1	1	0	
		1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	0
		1	1	0	0	1	0	1	0	1	1	1	1	1	1	1	0
Result	1	0	1	1	1	0	1	1	0	1	0	1	1	1	0	0	0

Step 3: convert the binary result to hexadecimal: $1\ 0111\ 0110\ 1011\ 1000_2$ = $176B8_{16}$

Method 2

The students **do not** need to know how to add 2 hexadecimal directly but here is the solution.

Carry	1	1	1	1	
		A	B	B	A
		C	A	F	E
Result	1	7	6	B	8

(The students will struggle with A+E but change to decimal in your head $10_{10}+14_{10} = 24_{10} = 18_{16}$.

With B+F+1 but change to decimal in your head $11_{10}+15_{10}+1_{10}=27_{10}= 1B_{16}$.

With B+A+1 but change to decimal in your head $11_{10}+10_{10}+1_{10}=22_{10}= 16_{16}$.

With A+C+1 but change to decimal in your head $10_{10}+12_{10}+1_{10}=23_{10}= 17_{16}$.)

An overflow has occurred, the result will NOT fit into 16 bits.