CS-117: Assignment 1

The Numerian Quest

Muhammad Mujtaba

CMD ID: 540040

mmujtaba.bese25seecs@seecs.edu.pk

Class: BESE 16B

Batch: 2k25

Number Systems:

Introduction:

Number systems are just a way to for expressing numbers in a consistent and determinable manner using a set amount of symbols.

When "N" numbers of unique symbols are used we call the number system a "Base N" number system.

Commonly used systems:

1. Decimal (Base 10):

We use this number system most commonly in our life, we have intuition for this number system.

In this number system we use 10 symbols:

0	1	2	3	4	5	6	7	8	9

We use this number system for general communication.

For *General Conversion* from $Base\ N\ to\ Base\ M$ we convert from $Base\ N\ to\ Base\ 10$ and then from $Base\ 10\ to\ Base\ M$.

2. Binary (Base 2):

This number system is used by computer. As computer treats electrical signal as either on (1) or off (0) this number system is extremely important because it allows humans to perform binary arithmetic operations in $Base\ 2$ and convert the output to $Base\ 10$.

In this number system we use 2 symbols:

0	1

Conversion from binary to decimal and vice versa are explained in *Special Conversion* Section.

1. Octal (Base 8):

This number system was created to visualize computer bytes in a more readable form. This system was used as a shorthand representation of binary numbers because each group of three binary digits (bits) can be represented by one octal digit. This number system is used by UNIX operating systems today to represent file permissions.

In this number system we use 8 symbols:

0	1	2	3	4	5	6	7

Conversion from octal to decimal and vice versa are explained in *Special Conversion* Section.

2. Hexadecimal (Base 16):

This number system is the modern standard to represent binary number. In hexadecimal instead of 3 bits to represent one octal number, a hexadecimal number can represent a group of 4 bits. This is very useful as computer operates on a byte (a group of 8 bits) of data at a time, so using 2 hexadecimal digits we can represent 1 complete byte. This number system is most used to represent computer memory or memory related things.

In this number system we use 16 symbols:

0	1	2	3	4	5	6	7	8	9
A*	B*	C*	D*	E*	F*				

** Can be upper or lower case*

Conversion from hexadecimal to decimal and vice versa are explained in *Special Conversion* Section.

Conversion:

General Conversion:

To convert from one arbitrary base N to another arbitrary base M:

1. Convert from Base N to Decimal

   Suppose Base N number, $d_k d_{k-1} \ldots d_1 d_0$

   Then

$$(number)_{10} = \sum_{i=0}^{k} d_i \times N^i$$

   Here:

   - $N =$ number of unique symbols in the base
   - $d =$ the value of the digit
   - $k =$ index from left

2. Convert from Decimal to Base M
   Apply Modulo operator until remainder is 0 or less than M, store them is a list as $[r_0,\ r_1,\ r_2,\ r_3, \ldots r_n]$.

   Then

$$(number)_M = r_n \ldots r_3 r_2 r_1 r_0$$

Special Conversion:

Binary to Decimal:

Binary numbers can be represented in decimal form as follows:

1. Suppose a binary number 10010.
2. To convert this number into decimal form we:

$$10010_2 = (1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0)_{10}$$

$$10010_2 = 18_{10}$$

3. To generalize, we add the following for all digits:

$$binary\_representation * 2^{(index\_from\_right - 1)}$$

Decimal to Binary:

Divide by 2 repeatedly and note the remainders. Suppose for number 18:

2	18	,	0
2	9	,	1
2	4	,	0
2	2	,	0
	1		

Since we need to write remainders in reverse order, $18_{10} = 10010_2$

Octal to Decimal:

Same as general, we multiply by 8^{index}.

Example:

$$22_8 = (2_8) * 8^1 + (2_8) * 8^0$$

$$22_8 = (2_{10}) * 8^1 + (2_{10}) * 8^0$$

$$22_8 = 18_{10}$$

Decimal to Octal:

1. First convert from decimal to binary
    a. Suppose 18_{10}, then in binary $18_{10} = 10010_2$
2. Separate into groups of three.
    a. $10010 \rightarrow 010 - 010$
3. Represent each group as an octal number.
    a. Groups are 010 and 010, in octal form 2 and 2 respectively.

4. Combine the numbers to get final answer.
    a. This $18_{10} = 22_8$

Hexadecimal to Decimal:

Same as general, we multiply by 16^{index}.

Example:

$$12_{16} = (1_{16}) * 16^1 + (2_{16}) * 16^0$$

$$12_{16} = (1_{10}) * 16^1 + (2_{10}) * 16^0$$

$$12_{16} = 18_{10}$$

Decimal to Hexadecimal:

1. First convert from decimal to binary
    a. Suppose 18_{10}, then in binary $18_{10} = 10010_2$
2. Separate into groups of four.
    a. $10010 \rightarrow 0001 - 0010$
3. Represent each group as a hexadecimal number.
    a. Groups are 001 and 0010, in hexadecimal form 1 and 2 respectively.
4. Combine the numbers to get final answer.
    a. This $18_{10} = 12_{16}$

Custom Number System:

For the assignment I used a number system of 32, with the following symbols.

0	1	2	3	4	5	6	7	8	9
a	b	c	d	e	f	g	h	i	j
k	l	m	n	o	p	q	r	s	t
u	v								

Custom to Decimal:

Same as general, we multiply by 32^{index}.

Example:

$$ok_{32} = (o_{32}) * 32^1 + (k_{32}) * 32^0$$

$$ok_{32} = (24_{10}) * 32^1 + (20_{10}) * 32^0$$

$$ok_{32} = 788_{10}$$

Decimal to Custom:

Same as general, we divide by 32, and note remainders, remainders in reverse are our numbers.

Example:

32	788	,	20
32	24	,	

These remainders are mapped as, $24 \rightarrow o$ and $20 \rightarrow k$

Thus

$$788_{10} = ok_{32}$$

Links:

Email: mmujtaba.bese25seecs@seecs.edu.pk

Github Repository Link: https://github.com/mmujtaba25/cs-117-assignment-1