25/09/2019 Daniel Hannon (19484286)

Assignment 1 – Numbering systems

There are many numbering systems in existence in the world today. This will cover a few of them. It will also show you how to use these numbering systems and how to add them and convert between number systems, I will assume for the rest that you know base 10

To start I will introduce the Numeration Systems that will be discussed throughout this work

Numeration Systems

Binary: any number is represented using a string of 0's and 1's and there positioning denotes a numeric power of 2^n

Hexedecimal: This extends the Base 10 Numbering system where numbers are represented by 0-9 and then extended with A-F this will be explained further in the content yet to follow. Each successive bit has a power of **V *** 16^n where **V** is the value. This will be explained further

Octal: This numbering system allows values of 0 – 7 to represent any number.

Positional Notation

Throughout this work I will use The positional Notation form a^n Where a is the base in which the number is written in and n is the distance from the least significant digit

Example:

453215 in Base 10 could be deconstructed as follows
$(4*10^5) + (5*10^4) + (3*10^3) + (2*10^2) + (1*10^1) + (5*10^0)$
within Base 10 this is not very significant but with other numbering systems it is useful for converting to base 10

Binary Numbers:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	1	0	1	0

(170 represented as binary)

How binary works is simple, 1 means to include the 2^n in your sum and 0 means you do not have to as a demonstration I will show you a step by step on how to represent 2492 in binary format

to start we need to figure out what the largest power of 2 is that fits into 2492, the answer is 2^{11} or 2048, for relatively small numbers this can be easily figured out.

$2492 – 2^{11} = 444$
$444 – 2^8 = 188$
$188 – 2^7 = 60$
$60 – 2^5 = 28$

$28 – 2^4 = 12$
$12 – 2^3 = 4$
$4 – 2^2 = 0$

From This we have formulated that you can represent 2492 as ($2^{11} + 2^8 + 2^7 + 2^5 + 2^4 + 2^3 + 2^2$) now in order to expand this into binary we need to represent 2^n values which are not present this is where the 0’s and 1’s come in

we use the 0’s to show numbers not used and 1’s to show numbers present so 2492 as a binary number would be

2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	0	1	1	0	1	1	1	1	0	0

100110111100

Now, Lets run through a random 10 bit binary number, let’s say 1100101100
first we break it down into the table shown above

2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	0	0	1	0	1	1	0	0

Then we add the values of 2^n that have a one in their respective position

$2^9 + 2^8 + 2^5 + 2^3 + 2^2 = 512 + 256 + 32 + 8 + 4 = 768 + 44 = 812$!!!

Octal

The Octal System is similar yet different instead of each number having a 0 or 1 to represent whether or not that specific number is used it is in the range 0 to 7 which denotes a multiplication of that specific power of 8^n where n is the distance from the least significant bit. The table below shows what each value in each respective row represents

*	8^2	8^1	8
0	0	0	0
1	64	8	1
2	128	16	2
3	192	24	3
4	256	32	4
5	320	40	5
6	384	48	6
7	448	56	7

Converting to Octal

Lets say that we want to convert a four digit decimal number to octal, as by referring to the table above we can deduce that the next column are multiples of 512

we will choose the number 2197 for this

unlike binary we will implement a new strategy for conversion. This consists of dividing value V by 8^n where n = distance from the least significant digit and then rounding down to the nearest whole number then subtracting 8^n * R from V and repeating where R is the result of the division

2197 / 8^3 rounded down is 4

2197 – 2048 = 149

149 / 8^2 rounded down is 2

149 – 128 = 27

27 / 8 rounbed down is 3

27 – 24 = 3

so 2197 is 4233 in octal

Converting from Octal to Decimal(Base 10)

Converting from Octal to Decimal is pretty easy once you have a solid grasp of what was setforth by the table above that being Sum($\mathbf{T_n * 8^{n-1)}}$ Where $\mathbf{T_n}$ Denotes the single digit value and n denotes the position and **n-1** denotes the position from the least significant digit of said number Lets Provide a working Example

6743 in Octal is....

$6 * 8^3 + 7 * 8^2 + 4 * 8 + 3$

6 * (512) + 7 * (64) + 32 + 3

3072 + 448 + 32 + 3

3520 + 35

3555 in Base 10!

Hexidecimal

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
16	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Hexidecimal follows the western numbering system of the leftmost digit denoting the Value of most significance and the rightmost denoting the least significant Digit. Each value follows the convention of representing $\mathbf{T_n*16^{n-1}}$ where n-1 represents distance from least significant digit and $\mathbf{T_n}$ indicates the value of the digit

Converting from base 10 to Hexidecimal

In order to convert a number from base 10 to Hexidecimal we're going to do something similar to converting to octal, the difference being that it is by 16^n rather than 8^n

Example Number

$3997 / 16^2 = 15.6$.......

3997 – 3840 = 157

157 / 16 = 9

157 – 144 = 13

So we have gotten the numbers

15, 9, and 13

Referring to the table above we can deduce that in Hexidecimal, 3997 is

F9D!!!

Converting Hexedecimal to base 10

Converting the other way is yet again like converting from octal

let's use the example

E0A6

when converted into respective digits it becomes

14 0 10 6

now we use the $\mathbf{T_n * 16^{n-1}}$ Function.

$14 * 16^3 + 160 + 6$

57344 + 166

57510

E0A6 becomes 57510 in base 10!

Advanced Methods

Converting between Types (Binary to Octal and Vice Versa)

now that we have covered the Binary and Octal systems, we should be fairly competent in converting from one to the other

Lets use two examples

1. Binary to Octal

Consider the number 01011010

first let’s split it in three sections in the format 2,3,3

01 011 010

why we do this is because each octal digit covers three bits of information as three bits have 8 possible states, just like each octal digit

using the table created earlier we can quickly deduce these in decimal form

1 3 2

As the maximum number that can be attained by three bits is 7 this is also the octal value of 01011010

2. Octal to Binary

consider the number 75634765

we simply perform the inverse of the previously defined operation

7	5	6	3	4	7	6	5
111	101	110	011	100	111	110	101

Then we merge them together and 75634765 becomes 111101110011100111110101 when converted from octal to binary!

Binary Addition/Subtraction

Binary addition has three straight forward rules
we only ever deal with two binary numbers at a time so if there happens to be three it is because there is a carry bit from the previous instance of addition

	1
+	0
0	1

	1
+	1
1	0

	1
	1
+	1
1	1

Lets try adding two random 8 digit binary numbers

10110101
01010101

	1	1	1		1		1	
	1	0	1	1	0	1	0	1
+	0	1	0	1	0	1	0	1
1	0	0	0	0	1	0	1	0

Subtracting two binary numbers

This is slightly more complex but still pretty straight forward, this is the only time a 2 will ever feature in binary.

	1	0
-	0	1
	0	1

	0	1
-	0	1
	0	0

Lets consider this subtraction

taking 01001110 from 11010011
Again, if there is a third bit present it is a carry bit

					1	2		
	1	1	0	1	0	0	1	1
-	0	1	0	0	1	1	1	0
	1	0	0	0	0	1	0	1