Binary
Decimal
Octal and
Hexadecimal number systems

A number can be represented with different base values. We are familiar with the numbers in the base 10 (known as decimal numbers), with digits taking values 0,1,2,…,8,9.

A computer uses a Binary number system which has a base 2 and digits can have only TWO values: 0 and 1.

A decimal number with a few digits can be expressed in binary form using a large number of digits. Thus the number 65 can be expressed in binary form as 1000001.

The binary form can be expressed more compactly by grouping 3 binary digits together to form an octal number. An octal number with base 8 makes use of the EIGHT digits 0,1,2,3,4,5,6 and 7.

A more compact representation is used by Hexadecimal representation which groups 4 binary digits together. It can make use of 16 digits, but since we have only 10 digits, the remaining 6 digits are made up of first 6 letters of the alphabet. Thus the hexadecimal base uses 0,1,2,….8,9,A,B,C,D,E,F as digits.

To summarize
Decimal : base 10
Binary : base 2
Octal: base 8
Hexadecimal : base 16

Decimal, Binary, Octal, and Hex Numbers

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Conversion of binary to decimal (base 2 to base 10)

Each position of binary digit can be replaced by an equivalent power of 2 as shown below.

2^{n-1}	2^{n-2}	……	……	2^3	2^2	2^1	2^0

Thus to convert any binary number replace each binary digit (bit) with its power and add up.

Example: convert $(1011)_2$ to its decimal equivalent

Represent the weight of each digit in the given number using the above table.

2^{n-1}	2^{n-2}	……	……	2^3	2^2	2^1	2^0
				1	0	1	1

Now add up all the powers after multiplying by the digit values, 0 or 1

$(1011)_2$

$= 2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 1$

$= 8 + 0 + 2 + 1$

$= 11$

Example2: convert $(1000100)_2$ to its decimal equivalent

$= 2^6 \times 1 + 2^5 \times 0 + 2^4 \times 0 + 2^3 \times 0 + 2^2 \times 1 + 2^1 \times 0 + 2^0 \times 0$

$= 64 + 0 + 0 + 0 + 4 + 0 + 0$

$= (68)_{10}$

Conversion of decimal to binary (base 10 to base 2)

Here we keep on dividing the number by 2 recursively till it reduces to zero. Then we print the remainders in reverse order.

Example: convert $(68)_{10}$ to binary

68/2 = 34 remainder is 0
34/ 2 = 17 remainder is 0
17 / 2 = 8 remainder is 1
8 / 2 = 4 remainder is 0
4 / 2 = 2 remainder is 0
2 / 2 = 1 remainder is 0
1 / 2 = 0 remainder is 1

We stop here as the number has been reduced to zero and collect the remainders in reverse order.
Answer = 1 0 0 0 1 0 0

Note: the answer is read from bottom (MSB, most significant bit) to top (LSB least significant bit) as $(1000100)_2$.

<u>You should be able to write a recursive function to convert a binary integer into its decimal equivalent.</u>

Conversion of binary fraction to decimal fraction

In a binary fraction, the position of each digit(bit) indicates its relative weight as was the case with the integer part, except the weights to in the reverse direction. Thus after the decimal point, the first digit (bit) has a weight of ½ , the next one has a weight of ¼ , followed by 1/8 and so on.

2^0 .	2^{-1}	2^{-2}	2^{-3}	2^{-4}	…	….	…
.	1	0	1	1	0	0	0

The decimal equivalent of this binary number 0.1011 can be worked out by considering the weight of each bit. Thus in this case it turns out to be

(1/2) x 1 + (1/4) x 0 + (1/8) x 1 + (1/16) x 1.

Conversion of decimal fraction to binary fraction

To convert a decimal fraction to its binary fraction, multiplication by 2 is carried out repetitively and the integer part of the result is saved and placed after the decimal point. The fractional part is taken and multiplied by 2. The process can be stopped any time after the desired accuracy has been achieved.

Example: convert $(0.68)_{10}$ to binary fraction.

0.68 * 2 = 1.36 integer part is 1
Take the fractional part and continue the process
0.36 * 2 = 0.72 integer part is 0
0.72 * 2 = 1.44 integer part is 1
0.44 * 2 = 0.88 integer part is 0

The digits are placed in the order in which they are generated, and not in the reverse order. Let us say we need the accuracy up to 4 decimal places. Here is the result.

Answer = 0. 1 0 1 0…..

Example: convert $(70.68)_{10}$ to binary equivalent.
First convert 70 into its binary form which is 1000110. Then convert 0.68 into binary form upto 4 decimal places to get 0.1010. Now put the two parts together.

Answer = 1 0 0 0 1 1 0 . 1 0 1 0….

Octal Number System

•Base or radix 8 number system.

•1 octal digit is equivalent to 3 bits.

•Octal numbers are 0 to7. (see the chart down below)

•Numbers are expressed as powers of 8. See this table

8^{n-1}	8^{n-2}	……	……	8^3	8^2	8^1	8^0
					6	3	2

Conversion of octal to decimal
(base 8 to base 10)

Example: convert $(632)_8$ to decimal

$= (6 \times 8^2) + (3 \times 8^1) + (2 \times 8^0)$

$= (6 \times 64) + (3 \times 8) + (2 \times 1)$

$= 384 + 24 + 2$

$= (410)_{10}$

Conversion of decimal to octal (base 10 to base 8)

Example: convert $(177)_{10}$ to octal equivalent

177 / 8 = 22 remainder is 1
22 / 8 = 2 remainder is 6
2 / 8 = 0 remainder is 2

Answer = 2 6 1

Note: the answer is read from bottom to top as $(261)_8$, the same as with the binary case.

Conversion of decimal fraction to octal fraction is carried out in the same manner as decimal to binary except that now the multiplication is carried out by 8.

Example: convert $(0.523)_{10}$ to octal equivalent up to 3 decimal places

0.523 x 8 = 4.184 ,its integer part is 4
0.184 x 8 = 1.472, its integer part is 1
0.472 x 8 = 3.776 , its integer part is 3

So the answer is $(0.413..)_8$

Conversion of decimal to binary (using octal)

When the numbers are large, conversion to binary would take a large number of division by 2. It can be simplified by first converting the number to octal and then converting each octal into its binary form:

Example: convert $(177)_{10}$ to its binary equivalent using octal form

Step 1: convert it to the octal form first as shown above

This yields $(2\ 6\ 1)_8$

Step 2: Now convert each octal code into its 3 bit binary form, thus 2 is replaced by 010, 6 is replaced by 110 and 1 is replaced by 001. The binary equivalent is $(010\ 110\ 001)_2$

Example: convert $(177.523)_{10}$ to its binary equivalent up to 6 decimal places using octal form.

Step 1: convert 177 to its octal form first, to get $(2\ 6\ 1)_8$ and then convert that to the binary form as shown above, which is $(010\ 110\ 001)_2$

Step 2: convert 0.523 to its octal form which is $(0.413..)_8$

Step 3: convert this into the binary form, digit by digit. This yields (0.100 001 011…)

Step 4: Now put it all together

$(010\ 110\ 001\ .\ 100\ 001\ 011\ldots)_2$

Conversion of binary to decimal (using octal)

First convert the binary number into its octal form. Conversion of binary numbers to octal simply requires grouping bits in the binary number into groups of three bits

•Groups are formed beginning with the Least Significant Bit and progressing to the MSB. Start from right hand side and proceed to left. If the left most group contains only a single digit or a double digit, add zeroes to make it 3 digits.

•Thus

$11\ 100\ 111_2$

$= 011\ \ 100\ \ 111_2$

$= \ \ 3\ \ \ 4\ \ \ 7_8$

And

$1\ 100\ 010\ 101\ 010\ 010\ 001_2$

$= 001\ 100\ 010\ 101\ 010\ 010\ 001_2$

$= 1425221_8$

Now it can be converted into the decimal form.

Hexadecimal Number System

•Base or radix 16 number system.

•1 hex digit is equivalent to 4 bits.

•Numbers are 0,1,2…..8,9, A, B, C, D, E, F.

B is 11, E is 14

•Numbers are expressed as powers of 16.

•$16^0 = 1$, $16^1 = 16$, $16^2 = 256$, $16^3 = 4096$, $16^4 = 65536$, …

Conversion of hex to decimal (base 16 to base 10)

Example: convert $(F4C)_{16}$ to decimal

$= (F \times 16^2) + (4 \times 16^1) + (C \times 16^0)$

$= (15 \times 256) + (4 \times 16) + (12 \times 1)$

Conversion of decimal to hex (base 10 to base 16)

Example: convert $(4768)_{10}$ to hex.

= 4768 / 16 = 298 remainder 0

= 298 / 16 = 18 remainder 10 (A)

= 18 / 16 = 1 remainder 2

= 1 / 16 = 0 remainder 1

Answer: 1 2 A 0

Note: the answer is read from bottom to top , same as with the binary case.

= 3840 + 64 + 12 + 0
= $(3916)_{10}$

Conversion of binary to hex

•Conversion of binary numbers to hex simply requires grouping bits in the binary numbers into groups of four bits.

•Groups are formed beginning with the LSB and progressing to the MSB.

•1110 0111_2 = $E7_{16}$

•1 1000 1010 1000 0111_2
= 0001 1000 1010 1000 0111_2
= 1 8 A 8 7_{16}