* NUMBER SYSTEMS *

* Number system is a way to represent numbers.

* The modern computers communicate and operate with binary numbers which use only digits 0 and 1.

* To represent long strings of bits in the addresses, instructions and data in digital systems, Octal and Hexadecimal number systems are used.

* The four number systems used in digital systems are
  1. Decimal Number system.
  2. Binary Number system.
  3. Octal Number system.
  4. Hexadecimal Number system.

* In any number system, the digits of a number has two parts. These two parts are integer and fractional part set apart by a radix point (.)

* In general, a number with a decimal point (.) is represented by a series of coefficients:

$$\boxed{N = \underbrace{a_{n-1}\, a_{n-2}\, a_{n-3} \cdots a_0}_{\text{Integer part}} \cdot \underbrace{a_{-1}\, a_{-2} \cdots a_{-m}}_{\text{fractional part}}}$$

Here, a_{n-1} is Most Significant Bit (MSB) & a_{-m} is Least Significant Bit (LSB)

Radix / Base :

* The Base / radix of a number system is defined as the number of unique digits used to represent a number (including 0).

* In any number system, a number expressed in a base-r system has coefficients multiplied by the powers of 'r'.

$$\boxed{a_n \cdot r^n + a_{n-1} \cdot r^{n-1} + \cdots a_2 \cdot r^2 + a_1 \cdot r^1 + a_0 \cdot r^0 + a_{-1} \cdot r^{-1} + a_{-2} \cdot r^{-2} + a_{-m} \cdot r^{-m}}$$

* To distinguish numbers of different bases, we enclose the coefficients in parantheses and write the base value.

Decimal Number System	Binary Number System	Octal Number System	Hexadecimal Number System.
Base/Radix (r) = 10	Base/Radix (r) = 2	Base/Radix (r) = 8	Base/ Radix (r) = 16
Range : 0 to 9	Range : 0 to 1	Range : 0 to 7	Range : 0 to 15

NUMBER BASE CONVERSIONS.

1. Any Radix to Decimal :

* Conversion of any base to decimal is done by expanding the number in a power series by multiplying each digit in the number by its positional weight and adding all the terms.

2. Decimal to Any Radix :

* Convert integer part by successive Division Method.
* Convert fractional part by successive Multiplication Method.

3. Binary to Octal :

* Group of 3 bits replaced by its decimal equivalent.

4. Binary to Hexadecimal :

* Group of 4 bits replaced by its decimal equivalent.

5. Octal to Binary :

* Write 3-bit binary number (421) equivalent to each octal digit and remove any leading and trailing zeros.

6. Hexadecimal to Binary :

* Write 4-bit binary number (8421) equivalent to each Hexadecimal digit and remove any leading and trailing zeros.

7. Octal to Hexadecimal :

* Octal → Binary → Hexadecimal

8. Hexadecimal to Octal :

* Hexadecimal → Binary → Octal.

9. Any Radix to Any Radix :

* Any Radix → Decimal → Another Radix.

① ANY RADIX TO DECIMAL NUMBER SYSTEM:

a) Binary to Decimal

$(1101.101)_2 = (?)_{10}$

Solution:

$$\begin{array}{cccccccc} 2^3 & 2^2 & 2^1 & 2^0 & & 2^{-1} & 2^{-2} & 2^{-3} \\ 1 & 1 & 0 & 1 & . & 1 & 0 & 1 \end{array}_2$$

$1 \times 2^{-3} = 0.125$

$0 \times 2^{-2} = 0.25$

$1 \times 2^{-1} = 0.5$

$1 \times 2^{0} = 1$

$0 \times 2^{1} = 0$

$1 \times 2^{2} = 4$

$1 \times 2^{3} = 8$

13.875

$\therefore \boxed{(1101.101)_2 = (13.875)_{10}}$

b) Octal to Decimal

$(475.25)_8 = (?)_{10}$

Solution:

$$\begin{array}{ccccccc} 8^2 & 8^1 & 8^0 & & 8^{-1} & 8^{-2} \\ 4 & 7 & 5 & . & 2 & 5 \end{array}$$

$5 \times 8^{-2} = 0.0781$

$2 \times 8^{-1} = 0.25$

$5 \times 8^{0} = 5$

$7 \times 8^{1} = 56$

$4 \times 8^{2} = 256$

317.3281

$\therefore \boxed{(475.25)_8 = (317.3281)_{10}}$

c) Hexadecimal to Decimal

$(9B2.1A)_{16} = (?)_{10}$

Solution:

$$\begin{array}{ccccccc} 16^2 & 16^1 & 16^0 & & 16^{-1} & 16^{-2} \\ 9 & B & 2 & . & 1 & A \end{array}$$

$10 \times 16^{-2} = 0.0391$

$1 \times 16^{-1} = 0.0625$

$2 \times 16^{0} = 2$

$11 \times 16^{1} = 176$

$9 \times 16^{2} = 2304$

2482.1016

$\therefore \boxed{(9B2.1A)_{16} = (2482.1016)_{10}}$

d) $(3102.12)_4 = (?)_{10}$

Solution:

$$\begin{array}{ccccccc} 4^3 & 4^2 & 4^1 & 4^0 & & 4^{-1} & 4^{-2} \\ 3 & 1 & 0 & 2 & . & 1 & 2 \end{array}$$

$2 \times 4^{-2} = 0.125$

$1 \times 4^{-1} = 0.25$

$2 \times 4^{0} = 2$

$0 \times 4^{1} = 0$

$1 \times 4^{2} = 16$

$3 \times 4^{3} = 192$

210.375

$\therefore \boxed{(3102.12)_4 = (210.375)_{10}}$

e) $(4021.2)_5 = (?)_{10}$

Solution:

$$\begin{array}{ccccccc} 5^3 & 5^2 & 5^1 & 5^0 & & 5^{-1} \\ 4 & 0 & 2 & 1 & . & 2 \end{array}$$

$2 \times 5^{-1} = 0.4$

$1 \times 5^{0} = 1$

$2 \times 5^{1} = 10$

$0 \times 5^{2} = 0$

$4 \times 5^{3} = 500$

511.4

$\therefore \boxed{(4021.2)_5 = (511.4)_{10}}$

② DECIMAL TO ANY RADIX

Example: $(5679.329)_{10}$ = a) $(?)_2$ b) $(?)_8$ c) $(?)_{16}$ d) $(?)_9$

Integer part : 5679, Fractional part : 329

5679 ↓ Successive Division Method

329 ↓ Successive Multiplication Method.

Solution:

a) $(5679.329)_{10} = (?)_2$

* Applying successive division on Integer part.

2	5679	
2	2839	-1
2	1419	-1
2	709	-1
2	354	-1
2	177	-0
2	88	-1
2	44	-0
2	22	-0
2	11	-0
2	5	-1
2	2	-1
	1	-0

$(5679)_{10} = (1011000101111)_2$.

* Applying successive multiplication on Fractional part.

$0.329 \times 2 = 0.648$
$0.648 \times 2 = 1.296$
$0.296 \times 2 = 0.592$
$0.592 \times 2 = 1.084$

$(0.329)_{10} = (0.0101)_2$

$\therefore \boxed{(5679.329)_{10} = (1011000101111.0101)_2}$

b) $(5689.329)_{10} = (?)_8$

* Applying successive division on Integer part

8	5689	
8	711	-1
8	88	-7
8	11	-0
	1	-3

$(5689)_{10} = (13071)_8$

* Applying successive multiplication on Fractional part.

$0.329 \times 8 = 2.632$
$0.632 \times 8 = 5.056$
$0.056 \times 8 = 0.448$
$0.448 \times 8 = 3.584$

$(0.329)_{10} = (0.2503)_8$

$\therefore \boxed{(5689.329)_{10} = (13071.2503)_8}$

c) $(5689.329)_{10} = (?)_{16}$

* Applying successive division on Integer part.

16	5689	
16	355	-9
16	22	-3
	1	-6

$(5689)_{10} = (1639)_{16}$

* Applying successive multiplication on Fractional part.

$0.329 \times 16 = 5.264$
$0.264 \times 16 = 4.224$
$0.224 \times 16 = 3.584$
$0.584 \times 16 = 9.344$

$(0.329)_{10} = (0.5439)_{16}$

$\therefore \boxed{(5689.329)_{10} = (1639.5439)_{16}}$

d) $(5689.329)_{10} = (?)_9$

* Applying successive division on Integer part

$$\begin{array}{r|l} 9 & 5689 \\ 9 & 632 \;-\; 1 \\ 9 & 70 \;-\; 2 \\ & 7 \;-\; 7 \end{array}$$

$(5689)_{10} = (7721)_9$

* Applying successive multiplication on fractional part.

$0.329 \times 9 = 2.961$
$0.961 \times 9 = 8.649$
$0.649 \times 9 = 5.841$
$0.841 \times 9 = 7.569$

$(0.329)_{10} = (2857)_9$.

$\therefore$ $\boxed{(5689.329)_{10} = (7721.2857)_9}$

③ BINARY TO OCTAL CONVERSION.

Step 1:

* For Integer part, make group of 3 bits starting from LSB
* for fractional part, make group of 3 bits starting from MSB

Step 2:

* Add 0's at the ends if required and convert 3 bits into decimal equivalent.

Example:

1) $(1001011.1011)_2 = (?)_8$

Solution:

001 | 001 | 011 . 101 | 100

1 1 3 . 5 4

$\therefore$ $\boxed{(1001011.1011)_2 = (113.54)_8}$

2) $(1101.11)_2 = (?)_8$

Solution:

001 | 101 . | 110 |

1 5 . 6

$\therefore$ $\boxed{(1101.11)_2 = (15.6)_8}$

④ BINARY TO HEXADECIMAL CONVERSION

Step 1:

* For integer part, make group of 4-bits starting from LSB
* For fractional part, make group of 4-bits starting from MSB.

Step 2:

* Add 0's at the ends if required and convert 4 bits into decimal equivalent.

Example

1) $(110100111.111101)_2 = (?)_{16}$

Solution:

00 1 | 1010 | 0111 . 1111 | 0100

1 A 7 . F 4

$\therefore \boxed{(110100111.111101)_2 = (1A7.F4)_{16}}$

2) $(1101101110.10011010)_2 = (?)_{16}$

Solution:

00 11 | 0110 | 1110 . 1001 | 1010

3 6 E . 9 A

$\therefore \boxed{(1101101110.10011010)_2 = (36E.9A)_{16}}$

(5) OCTAL TO BINARY CONVERSION

Step 1: Write the 3 bit binary equivalent to each octal digit.

Step 2: Remove any trailing and leading zeros.

Example

1) $(125.62)_8 = (?)_2$

Solution

1	2	5 .	6	2
001	010	101	110	010

$= (1010101.110010)_2$

$\therefore \boxed{(125.62)_8 = (1010101.110010)_2}$

2) $(634.27)_8 = (?)_2$

6	3	4 .	2	7
110	011	100	010	111

$= (110011100.010111)_2$

$\therefore \boxed{(634.27)_8 = (110011100.010111)_2}$

⑥ HEXADECIMAL TO BINARY CONVERSION

Step 1: Write the 4-bit binary equivalent to each hexadecimal digit.

Step 2: Remove any trailing and leading zeros.

Example

1) $(8A9.B4)_{16} = (?)_2$

Solution

8	A	9	.	B	4
1000	1010	1001		1011	0100

$= (100010101001.101101)_2$

$\therefore \boxed{(8A9.B4)_{16} = (100010101001.101101)_2}$

2) $(1FF3.00A)_{16} = (?)_2$

Solution

1	F	F	3	.	0	0	A
0001	1111	1111	0011		0000	0000	1010

$= (1111111110011.0000000001010)_2$

$\therefore \boxed{(1FF3.00A)_{16} = (1111111110011.0000000001010)_2}$

⑦ OCTAL TO HEXADECIMAL CONVERSION

Step 1: Convert each octal digit to its 3 digit binary equivalent.

Step 2: Now convert this binary number to hexadecimal by making group of 4-bits.

Step 3: Add zeros at the ends if required & write its hexadecimal equivalent

Example

1) $(615.25)_8 = (?)_{16}$

Solution:

octal → 6 1 5 . 2 5

Binary equivalent → 110 001 101 . 010 101

Converting Binary to Hexadecimal

0001	1000	1101 .	0101	0100
1	8	D .	5	4

$\therefore \boxed{(615.25)_8 = (18D.54)_{16}}$

2) $(167.32)_8 = (?)_{16}$

Solution:

octal → 167 . 32

Binary equivalent → 001 110 111 . 011 010

Converting Binary to Hexadecimal

0000	0111	0111 .	0110	1000
0	7	7 .	6	8

$\therefore \boxed{(167.32)_8 = (77.68)_{16}}$

⑧ HEXADECIMAL TO OCTAL CONVERSION

Step 1: Convert each hexadecimal digit into its 4-bit binary equivalent

Step 2: Convert this binary number into octal by making a group of 3 bits.

Step 3: Add trailing zeros at the ends if necessary and write its octal equivalent.

Example

① $(BC66.AF)_{16} = (?)_8$

Solution:

Hexadecimal → BC66.AF

Binary equivalent → 1011 1100 0110 0110 . 1010 1111

Converting Binary to Octal:

001	011	110	001	100	110 .	101	011	110
1	3	6	1	4	6 .	5	3	6

$\therefore$ $\boxed{(BC66.AF)_{16} = (136146.536)_8}$

② $(FF3.AB)_{16} = (?)_8$

Solution:

Hexadecimal → FF3.AB

Binary equivalent → 1111 1111 0011 . 1010 1011

Converting Binary to Octal:

111	111	110	011 .	101	010	110
7	7	6	3 .	5	2	6

$\therefore$ $\boxed{(FF3.AB)_{16} = (7763.526)_8}$

(9) ANY RADIX TO ANY OTHER RADIX

Step 1 : Convert the number in any radix to Decimal

Step 2 : Convert from Decimal to the other Radix.

Example

1) Convert $(432.34)_5$ to $(?)_6$

Step 1: Any Radix to Decimal

5^2	5^1	5^0		5^{-1}	5^{-2}
4	3	2	.	3	4

$4 \times 5^{-2} = 0.16$

$3 \times 5^{-1} = 0.6$

$2 \times 5^{0} = 2$

$3 \times 5^{1} = 15$

$4 \times 5^{2} = 100$

117.76

$\therefore (432.34)_5 = (117.76)_{10}$

Step 2: Decimal to Base 6

$(117.76)_{10} = (?)_6$

* Successive division on Integer part

$$\begin{array}{r|l} 6 & 117 \\ 6 & 19 \;\; -3 \\ & 3 \;\; -1 \end{array}$$

$(117)_{10} = (313)_6$

* Successive Multiplication on fractional part

$0.76 \times 6 = \boxed{4}.56$

$0.56 \times 6 = \boxed{3}.36$

$0.36 \times 6 = \boxed{2}.16$

$(0.76)_{10} = (0.432)_6$

$(117.76)_{10} = (313.432)_6$

$\therefore \boxed{(432.34)_5 = (117.76)_{10} = (313.432)_6}$

* * * *

2) Convert $(213.03)_4$ to $(?)_7$

Step 1: Any Radix to Decimal

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

$3 \times 4^{-2} = 0.187$

$0 \times 4^{-1} = 0$

$3 \times 4^{0} = 3$

$1 \times 4^{1} = 4$

$2 \times 4^{2} = 32$

39.187

$\therefore (213.03)_4 = (39.187)_{10}$

Step 2 : Decimal to Base 7

$(39.187)_{10} = (?)_7$

* Successive division on Integer part

$$\begin{array}{r|l} 7 & 39 \\ & 5 \;\; -4 \end{array}$$

$(39)_{10} = (54)_7$

* Successive Multiplication on fractional part

$0.187 \times 7 = \boxed{1}.309$

$0.309 \times 7 = \boxed{2}.163$

$0.163 \times 7 = \boxed{1}.141$

$(0.187)_{10} = (0.121)_7$

$(39.187)_{10} = (54.121)_7$

$\therefore \boxed{(213.03)_4 = (39.187)_{10} = (54.121)_7}$

* * *

* ARITHMETIC OPERATIONS *

* BINARY ADDITION *

Rules for Binary Addition

A	B	Sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

① Add $(1010)_2$ and $(111)_2$

Solution

$$\begin{array}{r} 1010 \\ +\ 111 \\ \hline 10001 \end{array}$$

$\therefore (1010)_2 + (111)_2 = (10001)_2$

② $1101.101 + 111.011 = ?$

Solution

$$\begin{array}{r} 1101.101 \\ 111.011 \\ \hline 10101.000 \end{array}$$

$\therefore (1101.101)_2 + (111.011)_2 = (10101.000)_2$

③ Add $(37)_{10}$ and $(56)_{10}$ in binary

Solution:

Convert decimal numbers to Binary

$$\begin{array}{l|l} 2 & 37 \\ 2 & 18 - 1 \\ 2 & 9 - 0 \\ 2 & 4 - 1 \\ 2 & 2 - 0 \\ & 1 - 0 \end{array} \qquad \begin{array}{l|l} 2 & 56 \\ 2 & 28 - 0 \\ 2 & 14 - 0 \\ 2 & 7 - 0 \\ 2 & 3 - 1 \\ & 1 - 1 \end{array}$$

$(37)_{10} = (100101)_2$ $\qquad$ $(56)_{10} = (111000)_2$

$(37)_{10}$ $\quad$ 100101

$(56)_{10}$ $\quad$ 111000

$\boxed{1011101}$ Ans.

To verify

$37 + 56 = 93.$

Result of Binary Addition = 1011101

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

$\therefore 64 + 0 + 16 + 8 + 4 + 0 + 1$

$= \underline{93}$

$\therefore (37)_{10} + (56)_{10} = (93)_{10}$

④ Add $1010.11_2 + 1101.10_2 + 1001.11_2 + 1111.11_2$

Solution:

$$\begin{array}{r} {\scriptstyle 10\ 10\ 10\ 10\ \ \ 1} \\ 1\ 0\ 1\ 0\ .\ 1\ 1 \\ 1\ 1\ 0\ 1\ .\ 1\ 0 \\ 1\ 0\ 0\ 1\ .\ 1\ 1 \\ 1\ 1\ 1\ 1\ .\ 1\ 1 \\ \hline (110001.11)_2 \end{array}$$ Ans.

$$\begin{array}{c} 1 \\ +1 \\ \hline 10 \end{array} \quad \begin{array}{c} 10 \\ +1 \\ \hline 11 \end{array} \quad \begin{array}{c} 11 \\ +1 \\ \hline 100 \end{array} \quad \begin{array}{c} 100 \\ +1 \\ \hline 101 \end{array} \quad \begin{array}{c} 101 \\ +1 \\ \hline 110 \end{array} \quad \begin{array}{c} 110 \\ +1 \\ \hline 111 \end{array} \quad \begin{array}{c} 111 \\ +1 \\ \hline 1000 \end{array}$$

* BINARY - MULTIPLICATION *

Rules for Binary Multiplication

A	B	A·B
0	0	0
0	1	0
1	0	0
1	1	1

① Multiply $(1101)_2$ with $(110)_2$.

```
      1101
    x  110
  -------
      0000
     1101
    1101
  -------
   1001110
```

$\therefore (1101)_2 \times (110)_2 = (1001110)_2$

② Multiply 1011.101 with 101.01

```
      1011.101
      x 101.01
  ----------
        1 10
      10 1011101
     0 0000000
    1 0111101
   0 0000000
  1011101
  ----------
  111101.00001
```

$\therefore (1011.101)_2 \times (101.01)_2 = (111101.00001)_2$

* BINARY DIVISION *

Rules for Binary Division

$$\boxed{\begin{array}{l} 0 \div 1 = 0 \\ 1 \div 1 = 1 \end{array}}$$

① Divide $(11011011)_2$ by $(110)_2$

```
        100100.1
110 ) 11011011
      110
      ------
       0110
        110
       -----
         0110
          110
         ----
            0
```

$\therefore (11011011) \div (110) = (100100.1)_2$

② $(110101.11)_2 \div (101)_2$.

```
        1010.11
101 ) 110101.11
      101
      ---
       110
       101
       ---
        111
        101
        ---
         101
         101
         ---
           0
```

0-0 = 0
0-1 = 1 1 borrow
1-0 = 1
1-1 = 0

$\therefore (110101.11)_2 \div (101)_2 = (1010.11)_2$

* BINARY SUBTRACTION *

Rules for Binary subtraction

A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

10 − 1 = 1

1) $(11101100)_2 - (00110010)_2$

```
  11101100
(-) 00110010
  --------
  10111010
```
Ans

$\therefore (11101100)_2 - (00110010)_2 = (10111010)_2$.

2) $(1110)_2 - (1101)_2$

```
  1110
(-) 1101
  ----
  0001
```
Ans.

$\therefore (1110)_2 - (1101)_2 = (0001)_2$

3) $(10001.01)_2 - (1110.10)_2$

```
  10001.01
(-)  1110.10
  --------
  00010.11
```
Ans

$\therefore (10001.01)_2 - (1110.10)_2 = (00010.11)_2$

4) $(1001.11)_2 - (1000.11)_2$

```
  1001.11
(-) 1000.11
  -------
  0001.00
```
Ans

$\therefore (1001.11)_2 - (1000.11)_2 = (0001.00)_2$

* COMPLEMENTS OF NUMBERS *

* In digital computers, complements are used to simplify the subtraction operation.
* There are two types of complements for each <u>base-r system</u>.
  1. The Radex complement (r's complement)
  2. Diminished radex complement (r-1's complement).
* Given a number N in base 'r' having n digits.

$\Rightarrow$ r-1 complement $= (r^n - 1) - N$

$\Rightarrow$ r's complement $= r^n - N$. (ie) (r-1 complement + 1)

* SUBTRACTION WITH COMPLEMENTS *

Rules for subtraction using Complements:

The subtraction of two n-digit numbers M-N in base 'r' can be done as follows:

1) Add the minuend M to the r's complement / (r-1)'s complement of the subtrahend N.

2) If $M \geq N$, the sum will produce an end carry.

→ In case of 1's and 9's complement, add end carry to sum which gives the final result. (ie r-1 complement)

→ In case of 2's and 10's Complement, discard the end carry. (i.e r complement)

3) If $M < N$, the sum does not produce an end carry. The result is the r's complement of (N-M).

→ To obtain the answer in a familiar form, take the r's complement of the sum and place a negative sign in front.

① Using 10's complement, subtract 72532 − 3250.

Solution:

Given M = 72532, N = 03250

Find 10's complement of N:

10's complement = 9's complement + 1

9's complement ⇒
```
  99999
  03250
  -----
  96749
```

10's complement ⇒
```
     +1
  -----
  96750
```

```
M =                 72532
10's complement (+) 96750
of N                -----
                    69282
```

End Carry is produced. In case of r's complement (10's complement) the carry can be discarded.

∴ 72532 − 3250 = 69282

② Using 10's complement, subtract 3250 − 72532.

Solution:

Given M = 03250, N = 72532

Find 10's complement of N:

10's complement = 9's complement + 1

9's complement ⇒
```
  99999
- 72532
  -----
  27467
```

10's complement ⇒
```
     +1
  -----
  27468
```

```
M =                 03250
10's comp (+)       27468
of N                -----
                    30718
```

There is no end carry.

∴ The answer is − (10's complement of 30718)

9's complement of 30718 =
```
  99999
  30718
  -----
  69281
```

10's complement =
```
     +1
  -----
  69282
```

∴ 3250 − 72532 = −69282

Finding Complements of Base-r numbers:

① Consider the Binary Number $N = (1011000)_2$. Find r's complement and (r-1)'s complement.

Solution:

Given a Binary number. Radix $r = 2$.

1's complement: (r-1 complement)

* It is found by subtracting each digit of N from 1

```
   1111111
(-) 1011000
   -------
   0100111  → 1's complement
```

(or) 1's complement of a Binary Number N is formed by changing 1's to 0's and 0's to 1's.

2's complement: (r complement)

2's complement = 1's complement + 1

```
   0100111
       + 1
   -------
   0101000  → 2's complement of N.
```

② Consider the Decimal Number $N = (546700)_{10}$. Find its r's complement and (r-1)'s complement.

Solution:

Given a Decimal Number. Radix $r = 10$.

9's complement: (r-1 complement)

* It is found by subtracting each digit of N from (r-1) (ie) 9

```
   999999
(-) 546700
   ------
   453299  → 9's complement
```

10's complement: (r complement)

10's complement = 9's complement + 1

```
   453299
+       1
   ------
   453300  → 10's complement.
```

③ Consider the Octal Number $N = (756214)_8$. Find its r's complement and (r-1)'s complement.

Solution:

Given the Octal Number. Radix $r = 8$.

7's complement: (r-1 complement)

* It is found by subtracting each digit of N from 7 (r-1)

```
   777777
(-) 756214
   ------
   021563  → 7's complement
```

8's complement: (r complement)

8's complement = 7's complement + 1

```
   021563
       + 1
   ------
   021564  → 8's complement.
```

④ Consider a Hexadecimal Number $N = (5AF624)_{16}$. Find its r's complement and (r-1)'s complement.

Solution:

Given Hexadecimal Number. Radix $r = 16$

15's complement: (r-1 complement)

* It is found by subtracting each digit of N from 15 (F)

```
   FFFFFF  (15)
(-) 5AF624
   ------
   A509DB  → 15's complement
```

16's complement: (r's complement)

16's complement = 15's complement + 1

```
   A509DB
+       1
   ------
   A509DC  → 16's complement
```

*** Similarly r's complement and r-1's complement can be found for Numbers of Any base ***.

③ Given two binary numbers $X = 1010100$ and $Y = 1000011$, Perform the subtraction a) $X - Y$ and b) $Y - X$ using 2's complements.

Solution:

a) $X - Y$

Given $X = 1010100$, $Y = 1000011$

Find 2's complement of Y

2's complement = 1's complement + 1

1's complement of $Y = 0111100$

2's complement of $Y = 0111100 + 1 = 0111101$

$$\begin{array}{rr} X = & 1010100 \\ \text{2's complement of } Y = & (+)\ 0111101 \\ \hline & \boxed{1}0010001 \end{array}$$

End carry is produced. In case of r's complement (2's complement here), the carry can be discarded from the sum.

$\therefore (1010100)_2 - (1000011)_2 = (0010001)_2$.

b) $Y - X$

Find 2's complement of X

2's complement = 1's complement + 1

1's complement of $X = 0101011$

2's complement of $X = 0101011 + 1 = 0101100$

$$\begin{array}{rr} Y = & 1000011 \\ \text{2's comple of } X = & (+)\ 0101100 \\ \hline & 1101111 \end{array}$$

No End Carry. $Y - X = -$(2's complement of 1101111)

2's Complement of sum $= 0010000 + 1$

④ Given two binary Numbers $X = 1010100$ and $Y = 1000011$. Perform a) $X - Y$ and b) $Y - X$ using 1's complement

Solution:

a) $X - Y$

Given $X = 1010100$, $Y = 1000011$

Find 1's complement of Y

1's complement $= 0111100$

$$\begin{array}{rr} X = & 1010100 \\ \text{1's comp of } Y = & (+)\ 0111100 \\ \hline & \boxed{1}\ 0010000 \end{array}$$

End carry is produced. In case of (r-1)'s complement (1's complement here), the end carry is added to the sum to produce final result

$$\begin{array}{rr} \text{Sum} = & 0010000 \\ \text{End carry} = & 1 \\ \hline \end{array}$$

$\therefore X - Y = (0010001)_2$ Ans

b) $Y - X$

Find 1's complement of X

1's complement of $X = 0101011$

$$\begin{array}{rr} Y = & 1000011 \\ \text{1's comp of } X = & (+)\ 0101011 \\ \hline & 1101110 \end{array}$$

No End Carry. $Y - X = -$(1's complement of 1101110)

$\therefore Y - X = -(0010001)_2$ Ans.

* SIGNED BINARY NUMBERS *

* The signed numbers has a symbol (+ or −) or a bit (0 or 1) indicating the sign.

* Positive number start with 0. Eg: +5 = 0000 0101 (sign bit)
* Negative number start with 1. Eg −5 = 1000 0101
* The leftmost bit is the sign bit for signed Binary Numbers.

Examples:

① (+6) + (+13)

```
+6        0000 0110
+13   (+) 0000 1101
+19       0001 0011   = (19)10.
```

② (−6) + (+13)

Find 2's complement of −6

Binary of −6 = 1000 0110
1's complement = 1111 1001
2's complement = +1
1111 1010

```
−6        1111 1010
+13       0000 1101 (+)
+7   ✗    0000 0111   = (7)10.
```

③ (+6) + (−13)

Find 2's complement of −13

Binary of −13 = 1000 1101
1's complement = 1111 0010
2's complement = +1
1111 0011

```
+6        0000 0110
−13   (+) 1111 0011
−7        1111 1001
```

∴ $\boxed{(11111001)_2 = \text{2's complement of } +7}$

④ (−6) + (−13)

Find 2's complement of −6

Binary of −6 = 1000 0110
1's complement = 1111 1001
2's complement = +1
1111 1010

Find 2's complement of −13

Binary of −13 = 1000 1101
1's complement = 1110 0010
2's complement = +1
1110 0011

```
−6        1111 1010
−13   (+) 1110 0011
−19       1101 1101   = (−19)
```

∴ $\boxed{(11111101)_2 = \text{2's complement of } +19}$

Verify: 0001 0011 (2^4 2^1 2^0) ⇒ Binary of 19

1's complement of 19 = 1110 1100
2's complement of 19 = +1
−19 = 1110 1101

∴ (−6) + (−13) = $(11101101)_2$

* BINARY CODES *

* A Binary code represents a coding system using the binary digits 0 and 1 to represent a letter, digit or a character in an electronic device.

* "A n-bit binary code is a group of n-bits that assumes upto 2^n distinct combinations of 1's and 0's, with each combination representing one element of the set that is being coded."

* Binary codes are broadly classified into

1. Numeric codes
   i) Weighted codes. (BCD 8421, 5421, 2421, 84-2-1)
   ii) Non-weighted codes (Excess-3, Gray code)
2. Alphanumeric codes (ASCII)
3. Error detecting codes.

① Binary - Coded Decimal (BCD) code: (Weighted Code)

* Although the binary number system is the natural system for a computer, most of us are more accustomed to the decimal system.

* The difference is resolved by converting decimal numbers (0 to 9) to binary, perform all arithmetic calculations in binary and then convert the binary results back to decimal.

* BCD is a Weighted Numeric code. It is the most commonly used code.

* BCD is a code in which each digit of a decimal number is represented by a seperate group of 4-bits.

Decimal Digit	BCD 8 4 2 1	5 4 2 1	2 4 2 1	8 4 -2 -1
0	0000	0000	0000	0000
1	0001	1001	0001	0111
2	0010	0010	0010	0110
3	0011	0011	0011	0101
4	0100	0100	0100	0100
5	0101	1000	1011	1011
6	0110	1001	1100	1010
7	0111	1010	1101	1001
8	1000	1011	1110	1000
9	1001	1100	1111	1111

* BCD is a weighted code in which each bit position is assigned a weighting factor in such a way that each digit can be evaluated by adding the weights of all the 1's in the coded combination.

5421 code:

* The bit combination 0101, when weighted by 5421 code is:

$5\times0 + 4\times1 + 2\times0 + 1\times1 = 0+4+1 = 5$ (decimal equivalent)

* Some digits can be coded in two possible ways in the 5421 code.

Eg: Decimal 6 can be assigned to bit combination 1001 or 0110

2421 code:

* The bit combination 1101, when weighted by 2421 code is

$= 2\times1 + 4\times1 + 2\times0 + 1\times0 = 2+4 = 6$ (decimal equivalent)

* Some digits can be coded in two possible ways in 2421 code.

Eg: Decimal 4 can be assigned to bit combination 0100 or 1010.

84-2-1 code:

* 84-2-1 code assigns both positive and negative weights to a decimal code.

* The bit combination 0110 is interpreted as decimal 2 in 84-2-1

$8\times0 + 4\times1 + (-2)\times1 + -1\times0 = 4-2 = 2.$

8421 code: (commonly used BCD code)

* The decimal equivalent for 0110 in 8421 code is calculated as

$8\times0 + 4\times1 + 2\times1 + 1\times0 = 4+2 = 6.$

(2) Excess-3 Codes (Unweighted Codes)

* unweighted code (*)
* Each coded combination is obtained as corresponding binary value + 3.

Decimal	Binary	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

③ Gray Code (Unweighted Code)

* In gray code, only one bit changes when going from one number to the next.
* It is convenient to use the gray code in some types of analog to digital converters.

Decimal	Gray code
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000

④ ASCII Character Code:

* The alphanumeric codes represents alphabetic and numeric information as a series of 1's and 0's.
* The standard binary code for the alphanumeric characters is American Standard Code for Information Interchange (ASCII).
* ASCII is a 7-bit code and so 2^7 or 128 characters can be represented.
* The 7-bit code is used to represent 128 alphanumeric characters (i.e) 26 upper case letters, 26 lower case letters, 10 numerals (0-9), 32 printable characters and 34 control characters.

⑤ Error Detecting Code:

* When the digital information in the binary form is transmitted from one system to another, an error may occur.
* This means a signal corresponding to 0 may change to 1 or vice versa due to the presence of noise.

Parity Bit:

* The simplest technique for detecting error is to add an extra bit known as parity bit to indicate the total number of 1's is odd or even

Example:

Consider the following ASCII values and their odd and even parity.

	With even parity	With odd parity
ASCII = 1000001	01000001	11000001
ASCII = 1010100	11010100	01010100

* The eight-bit characters that include parity bits are transmitted to their destination.

* The parity of each character is then checked at the receiving end.

* If the parity of the received character is not even, then atleast one bit has changed in value during the transmission is detected.

* If an error is detected, the receiver may request for retransmission of the message by sending back ASCII NAK (-ve Ack)

* If no error is detected, the receiver sends back an ACK (acknowledgement) control character.

* BCD ADDITION *

* It is a numeric code in which each digit of a decimal number is represented by seperate group of bits.

Example:

Binary value of 18 : $\overset{2^4}{1}00\overset{2^1}{1}\overset{2^0}{0}$

BCD Value of 18 : 0001 1000

Procedure for BCD Addition:

Step 1: Write BCD equivalent for every decimal digit and perform addition using ordinary Addition rules.

Step 2: a) If sum <= 9 & No carry, the sum is in valid BCD form

b) If sum >= 10 & No carry, the result is invalid BCD digit. So, add 6 (0110) to get valid BCD digit and a BCD carry.

occurs when sum >= 16 ← c) If sum <= 9, with carry, the result is invalid and needs correction because of the carry. So, add 6 (0110) to get the valid BCD sum and a BCD carry.

Examples for BCD Addition:

① 4+5

Solution:

```
  4        0100
+ 5      + 0101
  9        1001   Ans BCD sum
           (9)
```

② 4+8

Solution:

```
  4        0100
+ 8      + 1000
 12        1100   Binary sum > 9
         + 0110   Add 6
    0001   0010   Ans. BCD sum
      1      2
```

③ 8+9

Solution

```
  8        1000
+ 9      + 1001
 17        10001  Binary sum < 9, with carry
         + 0110   Add 6
    0001 0111     Ans BCD sum
      1    7
```

④ Using BCD Addition, add 24+18

Solution:

```
                 1
  24     0010 | 0100
+ 18   + 0001 ↓ | 1000 ↓
  42     0100 | 1100   Binary sum > 9
       +      | 0110   Add 6
         0100 | 0010   Ans BCD sum
          4   |  2
```

⑤ Using BCD, add 184+576

Solution:

```
                 1        1
 184     0001 | 1000 | 0100
+576   + 0101 ↓ | 0111 ↓ | 0110 ↓
 760     0101 | 10000 | 1010   sum > 9
       +      | 0110  | 0110   Add 6
Ans BCD sum = 0111 | 0110 | 0000
                7    |  6   |  0
```

⑥ Using BCD, add 48+58

Solution:

```
                 1
 48      0100 | 1000
+58    + 0101 ↓ | 1000 ↓
 106     1010 | 10000   Binary sum < 9, with carry
       +  0110 | 0110   Add 6
Ans 0001  0000 | 0110   BCD sum.
      1     0   |  6
```

⑦ Using BCD, add 175+326

Solution:

```
                 1        1
 175     0001 | 0111 | 0101
+326   + 0011 ↓ | 0010 ↓ | 0110 ↓
 501     0101 | 1010 | 1011   Binary sum > 9,
       +      | 0110 | 0110   Add 6
Ans    0101   | 0000 | 0001   BCD sum
         5    |  0   |  1
```

⑧ Using BCD, add 589+199

Solution:

```
                 1        1
 589     0101 | 1000 | 1001
+199   + 0001 ↓ | 1001 ↓ | 1001 ↓
 788     0111 | 1 0010 | 1 0010   Sum < 9, carry
       +      | 0110   | 0110   Add 6
Ans    0111   | 1000   | 1000   BCD sum
         7    |  8   |  8
```