UNIT - I

1.1 DATA REPRESENTATION

DATA TYPES

- A data type is a type of data.
- Data type is a data storage format that can contain a specific type or range of values.
- Computer programs store data in variables
- Each variable must be assigned a specific data type.
- Some common data types include integers, floating point numbers, characters, strings, and arrays.

1.1.1 NUMBER SYSTEM

- A number system of base (also called radix) r is a system, which have r distinct sy mbols for r digits.
- A number is represented by a string of these symbolic digits.

Number system can be divided into three types

They are:

- ➢ Binary number
- ➢ Octal number
- ➢ Decimal number
- ➢ Hexa decimal number

1.2.2 DECIMAL, BINARY, OCTAL, HEXADECIMAL & CONVERSION

1. DECMAL NUMBER SYSTEM

- Decimal system is the most commonly used number system.
- The decimal system has a base or radix of 10.
- The 10 separate symbols namely 0,1,2,3,4,5,6,7,8 and 9.
- Also called Arabic numerals.
- The digit of a number system is a symbol, which represents an integral quantity.
- The base or radix of a number system is defined as the number of different digits, which
  can occur in each position in the number system.

2. BINARY NUMBER SYSTEM

- Base is 2.
- Two digits symbols are used : 0 and 1.
- The numbers in binary system are represented as combinations of these two sym bols.
- The binary system uses powers of 2.
- To distinguish between different radix numbers, the digits will be enclosed in parenthesis and radix is inserted as a subscript.
- The binary digit is also referred to as Bit

- A string of 4 bits is called a nibble
- a string of 8 bits is called a byte.
- A byte is the basic unit of data in computers.

The number 125 actually means 1* 10^2 + 2* 10^1 +5*10^0 .

3.OCTAL NUMBER SYSTEM

- The octal number system has a base or radix as 8
- Eight different symbols are used torepresent numbers.
- These are commonly 0,1,2,3,4,5,6,7.
- The first 20 octal numbers and their decimal equivalents in the table

Decimal	Octal	Decimal	Octal
0	0	11	13
1	1	12	14
2	2	13	15
3	3	14	16
4	4	15	17
5	5	16	20
6	6	17	21
7	7	18	22
8	10	19	23
9	11	20	24
10	12	21	25

First 20 Octal Numbers

4. HEXADECIMAL NUMBER SYSTEM

- The hexa decimal number system has a base or radix as 16
- 16 possible different numbers can be represented, the digits 0 through 9 and letters A, B,
- C, D, E and F are also used.
- Hexadecimal numbers are strings of these digits.
- The numbers in decimal, binary and hexadecimal is shown in the table.

Binary	Hexadecimal	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

First 16 Hexadecimal Numbers

CONVERSIONS

1. BINARY NUMBER TO DECIMAL NUMBER

1. $(1011)_2$

$=1*2^3 +0*2^2+1*2^1 +1*2^0$

$= 1*2*2*2+0*2*2+1*2+1*1$

$=8+ 0 + 2 + 1$

$=11$

$(1011)_2 =(11)_{10}$

2. $(10001)_2$ $=1*2^4+0*2^3+ 0*2^2+0*2^1 + 1*2^0$

$= 1*2*2*2*2+0*2*2*2+0*2*2+0*2+1*1$

$= 16 + 0 + 0 + 0 + 1$

$=17$

$(10001)_2 =(17)_{10}$

2. BINARY NUMBER TO OCTAL NUMBER

There is a simple trick for converting a binary number to an octal number. Simply group the binary digits into groups of 3, starting at the octal point,

BINARY	**OCTAL**
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

First 8 Octal Numbers

Let us convert the binary number 011101 into octal.

011101 = 011 101
= 3 5
= $(3\ 5)_8$

3. BINARY NUMBER TO HEXADECIMAL NUMBER

To convert binary to hexadecimal, a binary number into groups of four digits and convert each group of four digits according to the preceding code.

Here are some

Example:1

$(10111011)_2$ = 1011 1011
= B B

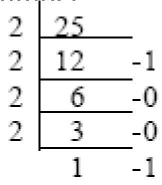
$(10111011)_2$= **$(B\ B)_{16}$**

Example:2

$(10010101)_2$ = 1001 0101
= 9 5
$(10010101)_2$ = **$(9\ \ 5)_{16}$**

4. DECIMAL NUMBER TO BINARY NUMBER

To convert a decimal number into binary number it requires successive division by 2writing down each quotient and its remainder. The remainders are taken in the reverse order,which is the binary equivalent of the decimal number

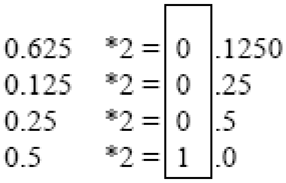


```
2 | 25
2 | 12   -1
2 |  6   -0
2 |  3   -0
     1   -1
```

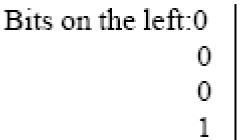
The binary equivalent for **25_{10} =11001_2**

To convert decimal fractions into equivalent binary fractions repeatedly double the decimal fraction.

The number (0 or 1) that appears on the left is written separately. The bits that are written in this manner are read from top to bottom with a decimal point on the left. For example if the given number is 0.0625 conversion is done in the following manner.



0.625	*2 =	0	.1250
0.125	*2 =	0	.25
0.25	*2 =	0	.5
0.5	*2 =	1	.0

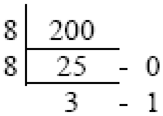


Bits on the left:0
0
0
1

Therefore, 0.0625_{10} = **$.0001_{2}$**

5. DECIMAL NUMBER TO OCTAL NUMBER

Conversion from decimal to octal can be performed by repeatedly dividing the decimal number by 8
and using each remainder as a digit in the octal number being formed.

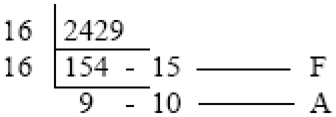


```
8 | 200
8 | 25  - 0
     3  - 1
```

Therefore $(200)_{10}$ = **$(310)_{8}$**

6. DECIMAL NUMBER TO HEXADECIMAL NUMBER

One way to convert decimal to Hexadecimal is the hex dabbles. The idea is as divide successively by
16, writing down the remainders.



```
16 | 2429
16 | 154 - 15 ——— F
      9  - 10 ——— A
```

Therefore $(2429)_{10}$ = **$(9AF)_{16}$**

7. OCTAL NUMBER TO DECIMAL NUMBER

To convert an octal number to a decimal number, we use the same sort of polynomial as was used in
the binary case, except that we now have a radix of 8 instead of 2.

Example: $(1213)_{8}$

$=1*8^3+2*8^2+1*8^1+3*8^0$

$=512 + 128 + 8 + 3$

$=651$

Therefore $(1213)_8$=**$(651)_{10}$**

8. OCTAL NUMBER TO BINARY NUMBER

The conversion from octal number to binary number is easily accomplished. Each octa l bit is converted to its three digit binary equivalent.

Example: 1
$(26153.7406)_8$

$(2 \quad 6 \quad 1 \quad 5 \quad 3 \ . \quad 7 \quad 4 \quad 0 \quad 6)_8$

$(010 \quad 110 \quad 001 \quad 101 \quad 011 \quad . \quad 111 \quad 100 \quad 000 \quad 110)_2$

Therefore $(26153.7406)_8 = (010\ 110\ 001\ 101\ 011\ .111\ 100\ 000\ 110)_2$

9. OCTAL NUMBER TO HEXADECIMAL NUMBER

The method of converting octal to hexadecimal number is to convert the given octal number to
binary number and then arrange the binary digits into groups of 4 starting at the binary point.

Example :1 Convert octal number 714.06 to hexadecimal.

$(714.06)_8$ $=(111\ 001\ 100.000\ 110)_2$
=0001 1100 1100 . 0001 1000
= 1 C C . 1 8

The hexadecimal equivalent of $(714.06)_8$ is **$(1CC.18)_{16}$**

10. HEXADECIMAL NUMBER TO BINARY NUMBER

To convert a hexadecimal number to a binary number, convert each hexadecimal digit to its 4-bit
equivalent using the code.
Example :1

$(\ 3 \quad 0 \quad 6 \quad . \quad D)_{16}$

$(\ 0011\ 0000\ 0110\ .\ 1101)_2$
9 A F

Therefore $(306.\ D)_{16}$ = **$(9AF)_2$**

Example :2
As another example, C5E2
C 5 E 2

1100 0101 1100 0010

Therefore $(C5E2)_{16}$ = **$(1100\ 0101\ \ 1100\ \ \ 0010)_2$**

11. HEXADECIMAL NUMBER TO OCTAL NUMBER

The conversion of Hexadecimal number to octal number involves two steps. First the method suggests to go from hexadecimal to binary numbers and second from binary to octal numbers Convert the hexadecimal into binary by writing 4 bits binary value for each bit in hexadecimal number and then arrange the binary digits into groups of three starting at the binary point.

Example:1

Convert Hexadecimal (1E.C) to octal conversion

(1E.C)16 = (0001 1110.1100)2
= (011 110.110)
=36.6

The octal equivalent of $(1E.C)_{16}$ is $(36.6)_{8}$

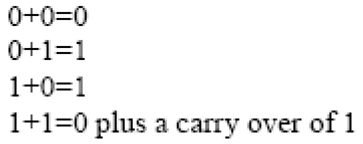
12. HEXADECIMAL NUMBER TO DECIMAL NUMBER

The conversion of Hexadecimal to decimal is straightforward but time consuming. In Hexadecimal number system each digit position corresponds to a power of 16. The weights of the digit positions in a hexadecimal number are as follows: For instance, BB represents Example:1

$$\begin{aligned} BB &= B*16^{1} + B*16^{0} \\ &= 11*16+11*1 \\ &= 176+11 \\ &= 187 \end{aligned}$$

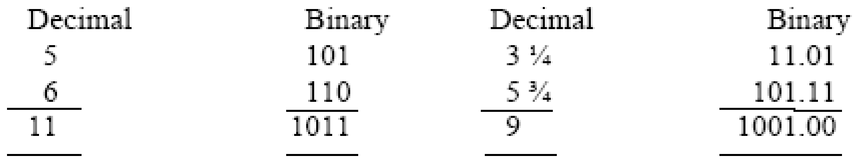
BINARY ADDITION AND SUBTRACTION

Binary addition is performed in the same manner as decimal addition. The complete table for binary addition is as follows:



0+0=0
0+1=1
1+0=1
1+1=0 plus a carry over of 1

'Carry over' are performed in the same manner as in decimal arithmetic. Since 1 is the largest digit in the binary system, any sum greater than 1 requires that a digit be carried. Examples:

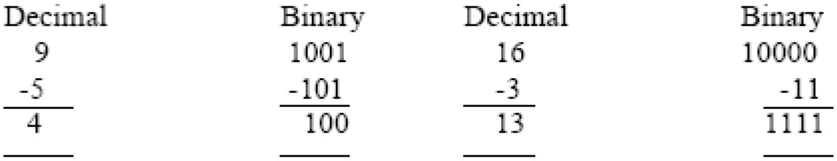


Decimal	Binary	Decimal	Binary
5	101	3 ¼	11.01
6	110	5 ¾	101.11
11	1011	9	1001.00

Subtraction is the inverse operation of addition. To subtract, it is necessary to establish a procedure for subtracting a larger from a smaller digit. The only case in which this occurs with binary numbers is when 1 is subtracted from 0. It is necessary to borrow 1 from the next column to the left. This is the binary subtraction table.

0-0=0
1-0=1
0-1=1 with a borrow of 1
1-1=0

Examples:



Decimal	Binary	Decimal	Binary
9	1001	16	10000
-5	-101	-3	-11
4	100	13	1111

1.1.3 DECIMAL

BINARY CODE:

- A binary coed is a group of n bits.
- A set of four elements can be codeed by a 2 bit code with any one of following bit combination: 00 01 10 11.

BINARY CODED DECIMAL

- Most popular decimal code
- Bcd is different from converting a decimal number to binary number

1.1.4 APLHANUMERIC REPRESENTATION

Character

- An alphanumeric character set is a set of elememts that includes the 10 decimal digits, 26 letters of the alphabet and number of special character such as +,- etc.
- A set contains between 32 and 64 elements with both upper & lower case.

ASCII

- Standard alphanumeric binary code is the ASCII(american standards code for information interchange)
- Binary codes are required since registers can hold binary information only.

1.1.5 COMPLEMENTS

- Complements are used to represent negative numbers in digital computers for simplifying the
- subtraction operation and logical manipulation.

There are two types of complements for each base –1 system.

1. The r's Complement
2. The (r-1)'s Complement

- When the value of the base is substituted the two types receive the names 2's and 1's complement for binary number or 10's and 9's complement for decimal numbers.
- The r's complement is sometimes called as "True Complement"
- The (r-1)'s complement as "Radix minus one's complement".

1. <u>The (r-1)'s complement</u>

- Given a positive number N in base r with an integer part of N digits and a fraction part of
- m digits, then (r-1)'s complements can be defined as $r^n - r^m - N$.
- The (r-1)'s complement in decimal system is 9's complement and 1's complement
- in case of binary.
- 9's complement of a decimal number is formed simply by subtracting every digit by 9.
- The 1's complement of a binary number is even simpler to form, the 1's are changed t o
- 0's and 0's are changed to 1's.

Example: The 9's complement of 52510 is

$(10^5 -1) –52510$

$=99999-52510$

$=47489$

The 9's complement of 0.3266 is

$=(1-10^{-4}) – 0.3266$

$=0.9999 – 0.3266$

$=0.6733$

The 9's complement of 25.638 is

$=(10^2 – 10^{-3}) –25.638$

$=99.999-25.638$

$=74.361$

Examples of 1's complement is as follows:

Example:

The 1's complement of 101100 is

$=(2^6 – 1)-101100$

$=(1000000 – 1) – 101100$

$=111111-101100$

$=010011$

The 1's complement of (0110) is

$= 1001$

2. <u>The r's complement</u>

- Given a positive number N in base r with an integer part of n digits the r's complemen t of

N is defined as r^n- N for N 0 and 0 for N=0.

- The r’s complement in decimal system is 10’s complement and 2’s complement in case of binary system.
- 2’s complement of 101101 is obtained from the 1’s complement 010010 by adding 1 to give 010011.
- 10’s complement can be formed by forming 9’s complement and then adding a 1 to the least significant digit.

1.1.6 9’S , 10’S , 1’S AND 2’S COMPLEMENT

Some numerical examples of 10’s complement is as follows:

Example:

The 10’s complement of (52510)10 is
=10^5 –52510
=47490

The 9’s complement of 326 is
=999-326
=673

The 10’s complement of 25.63810 is
=10^2 –25.638
=74.362

TWO'S COMPLEMENT

The two's complement of a binary number is obtained by:

1. Replacing all the 1's with 0's and the 0's with 1's. This is known as its one's complement.
2. Adding 1 to this number by the rules of binary addition.

Now you have the two's complement.

Example:

The decimal subtraction 29 - 7 = 22 is the same as adding (29) + (-7) = 22

1. Convert the number to be subtracted to its two's complement:

00000111	(decimal 7)
11111000	(one's complement)
+ 00000001	(add 1)
11111001	(two's complement)

2. 11111001 now represents -7.
3. Add

29	00011101
+- 7	11111001
22	(1)00010110

Note that the final carry 1 is ignored.

- The r’s complement can be obtained from the (r-

1)'s complement after the addition of r
to the least significant digit.

1.1.7 COMPLEMENTS ADDITION AND SUBTRACTION

1. Binary subtraction using 1's complement method:

Step1: Find 1's complement of the number to be subtracted(subtra hend).
Step2: Perform binary addition of this 1's complement number with the given first number(minuend).
Step3: Any end-around carry found, add to the least significant bit.

Example :

```
11001       11001
-10110   = + 01001
              1 00010 (end-carry is added to the least significant bit)
                    1
              ------------
      Ans: =  00011
            ------------
```

2. Binary subtraction using 2's complement method:

Step1: Find 2's complement of the number to be subtracted(subtra hend).
Step2: Perform binary addition of this 2's complement number with the given first number(minuend).
Step3: Any end-around carry found, drop it.

Example :

```
11011                11011
-11001            = +  00111
                  -----------------
                     1 00010 (carry is dropped).
                 ------------------
      Ans:  = 00010.
```

2's Complement Arithmetic

1. 47 + 23

Since 47 and 23 are not negative integers, we don't have to use two's complement for addition.

47 = (32 + 8 + 4 + 2 + 1) = 101111
23 = (16 + 4 + 2 + 1) = 10111

```
    1 1 1 1 1   carry row
    1 0 1 1 1 1
+     1 0 1 1 1
 ______________
  1 0 0 0 1 1
```

1000110 = 70

Notice that no sign bit is added in this instance, nor is the leftmost bit to be interpreted as a negative.

2. 56 + (-19)

56 = 111000 = 0111000   (Did you forget the sign bit?)
19 = 0010011

Two's complement of 0010011 = 1101101

```
   1 1 1      carry row
   0 1 1 1 0 0 0
 +  1 1 0 1 1 0 1
  1 0 1 0 0 1 0 1
```

After truncating the overflow '1', we get the positive number 0100101, which equals 37.

3. 72 + (-100)

72 = 01001000
100 = 01100100, and the two's complement of 01100100 = 10011100.

```
             1 1      carry row
   0 1 0 0 1 0 0 0
  + 1 0 0 1 1 1 0 0
   1 1 1 0 0 1 0 0
```

There is no overflow, and the leftmost digit is a 1, indicating that the result is a negative number.
First use 2's complement to convert 11100100 to 00011011, then convert 00011011 to decimal representation and take the negative. The answer is -28.

4. (-35) + (-58)

-35 = -00100011 = 11011101 (two's complement of 35)
-58 = -00111010 = 11000110 (two's complement of 58)

```
   1   1 1 1      carry row
   1 1 0 1 1 1 0 1
 +  1 1 0 0 0 1 1 0
 1 1 0 1 0 0 0 1 1
```

1.2 OTHER CODES:

1.2.1 GRAY CODE

- Digital systems can process data in discrete form only.
- Many physical systems supply
  continuous output data. The data must be converted into digital
  form before they can be used by a digital computer.
- Continuous, or analog information is converted into digital form by means of

an analog to digital converter.

- The reflected binary or gray code is shown in the table is sometimes used for the converted digital data.
- The advantage of the gray code over straight binary numbers is that the gray code changes
- by only one bit as it sequences from one number to the next.
- In other words, the change from any number to the next in sequence is recognized by a change of only one bit from 0 to 1 data is represented by the continuous change of a shaft position.
- The shaft is partitioned into segments with each segment assigned a number.

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

4 Bit Gray Code

1.2.2 BINARY CODED DECIMAL

In BCD number system a group of binary bit is used to represent each of 10 decimal digits. For instances, an obvious and natural code is a simple weighted binary code as shown in able.

BINARY CODE	**DECIMAL DIGIT**
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8

1001	9

Binary Coded Decimal

- This is known as a binary coded decimal 8421 code or simply BCD.
- Notice that 4 binarybits are required for each decimal digit and each bit is assigned a weight;
- for instance the rightmost bit has a weight of 1, and the leftmost bit in each
- code group has a weight of 8.
- By
  adding the weights of the positions in which 1's appear, the decimal digit represented by a code
- group may be derived.

For the decimal number 1246 to be represented, 16 bits are required:

1	2	4	6
0001	0010	0100	0110

Examples:

Convert decimal 4019 to BCD

4	0	1	9
0100	0000	0001	1001

The BCD equivalent of (4019) 10 is 0100 0000 0001 1001

Convert BCD number 0001 1001 0000 0111 to decimal

0001	1001	0000	0111
1	9	0	7

- BCD numbersare useful wherever decimal information is transferred into a computer .
- The pocket calculator is one of the best examples for the application of BCD numbers. Other examples of BCD system are electronic counters, digital voltmeter and digital clocks.

1.2.3 WEIGHTED CODE

- Each position of the number has a specific weight.
- Some example code are 8421, 2421, and 5211.
- Ith bits are multiplied by the weights indicated and the sum of the weighted bits ghives decimal digits
- **Example: 1101 using 2421 code**

= 2*1+4*1+2*0+1*1
=2+4+0+1
=7

1.2.4 EXCESS – 3 CODE

- The Excess-3 code is a decimal code that has been used in older computers.
- This is an un-weighted code.
- Its binary code assignment is obtained from the corresponding BCD equivalent
- binary number after the addition of binary 3 (0011).

DECIMAL	BCD	EXCESS 3 CODE
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
10	0001 0000	0100 0011
11	0001 0001	0100 0100

Excess – 3 code

1.2.5 EBCDIC CODE

- Extended BCD interchange code
- It is a alphanumeric code
- Same character symbols as ASCII but the bit assignment to characters are different
- It is used for business oriented purpose.

1.2.6 ERROR DETECTION AND ERROR CORRECTION

- An error detection code is a binary code that detects digital errors during transmission
- Detected errors cannot be corrected but there is indicated
- Common error detection code is used in parity bit
- Parity bit is an extra bit included with a binary message to make the total number of 1's either odd or even.
- If an additional bit is added to make the total number of 1's an odd number. Then it is called an odd parity
- The total number of 1's in the resulting code is even then it is called an even parity.
- Additional parity bit is normally placed on the left of the MSB
- Example:

Parity	Data	Odd parity total

bit		number
1	1010101	5
0	1101011	5
1	0100100	3
0	1010010	3

- Resulting code has one parity and seven data bits.
- It forms an eight bit word which is transmitted to the receiver.

Parity bit	Data	Even parity total number
1	1101011	6
0	1011111	6
1	1001001	4
0	1001000	2

- Example:
- 11010101 are received as 11010111 due to error because of some failure.