UNIT - 1

DATA REPRESENTATION

Dr. Saurabh Pal
Associate Professor & Head
Department of Computer Applications
Veer Bahadur Singh Purvanchal University, Jaunpur

OBJECTIVES

- Various types of number systems
- Conversion between number system
- Decimal to Binary, Octal, Hexadecimal Conversion
- Binary, Octal, Hexadecimal to Decimal Conversion

NUMBER SYSTEM

- Number systems are the technique to represent numbers in the computer system architecture, every value that you are saving or getting into/from computer memory has a defined number system.
- Computer architecture supports following number systems.
  - Binary number system
  - Octal number system
  - Decimal number system
  - Hexadecimal (hex) number system

DESCRIPTIONS OF NUMBER SYSTEM

- Binary Number System
  - A Binary number system has only two digits that are 0 and 1. Every number (value) represents with 0 and 1 in this number system. The base of binary number system is 2, because it has only two digits.
- Octal number system
  - Octal number system has only eight (8) digits from 0 to 7. Every number (value) represents with 0,1,2,3,4,5,6 and 7 in this number system. The base of octal number system is 8, because it has only 8 digits.

DESCRIPTIONS OF NUMBER SYSTEM

- Decimal number system
  - Decimal number system has only ten (10) digits from 0 to 9. Every number (value) represents with 0,1,2,3,4,5,6, 7,8 and 9 in this number system. The base of decimal number system is 10, because it has only 10 digits.
- Hexadecimal number system
  - A Hexadecimal number system has sixteen (16) alphanumeric values from 0 to 9 and A to F. Every number (value) represents with 0,1,2,3,4,5,6, 7,8,9,A,B,C,D,E and F in this number system. The base of hexadecimal number system is 16, because it has 16 alphanumeric values. Here A is 10, B is 11, C is 12, D is 13, E is 14 and F is 15.

DESCRIPTIONS OF NUMBER SYSTEM

Number system	Base	Used digits	Example
Binary	2	0,1	$(11110000)_2$
Octal	8	0,1,2,3,4,5,6,7	$(360)_8$
Decimal	10	0,1,2,3,4,5,6,7,8,9	$(240)_{10}$
Hexadecimal	16	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F	$(F0)_{16}$

NUMBER SYSTEM CONVERSION

- There are three types of conversion:
  - Decimal Number System to Other Base [for example: Decimal Number System to Binary Number System]
  - Other Base to Decimal Number System [for example: Binary Number System to Decimal Number System]
  - Other Base to Other Base [for example: Binary Number System to Hexadecimal Number System]

DECIMAL NUMBER SYSTEM TO OTHER BASE

- To convert Number system from Decimal Number System to Any Other Base is quite easy; you have to follow just two steps:
  - Divide the Number (Decimal Number) by the base of target base system (in which you want to convert the number: Binary (2), octal (8) and Hexadecimal (16)).
  - Write the remainder from step 1 as a Least Signification Bit (LSB) to Step last as a Most Significant Bit (MSB).

DECIMAL NUMBER SYSTEM TO BINARY

Decimal Number is : $(12345)_{10}$

2	12345	1	LSB
2	6172	0	
2	3086	0	
2	1543	1	
2	771	1	
2	385	1	
2	192	0	
2	96	0	
2	48	0	
2	24	0	
2	12	0	
2	6	0	
2	3	1	
	1	1	MSB

Binary Number is

$(11000000111001)_2$

DECIMAL NUMBER SYSTEM TO OCTAL

Decimal Number is : **(12345)₁₀**

8	12345		1	LSB
8	1543		7	
8	192		0	
8	24		0	
	3		3	MSB

Octal Number is

$(30071)_8$

DECIMAL NUMBER SYSTEM TO HEXADECIMAL

Decimal Number is : $(12345)_{10}$

16	12345	9	LSB
16	771	3	
16	48	0	
8	3	3	MSB

Hexadecimal Number is

$(3039)_{16}$

OTHER BASE SYSTEM TO DECIMAL

- To convert Number System from Any Other Base System to Decimal Number System, you have to follow just three steps:
  - Determine the base value of source Number System (that you want to convert), and also determine the position of digits from LSB (first digit's position – 0, second digit's position – 1 and so on).
  - Multiply each digit with its corresponding multiplication of position value and Base of Source Number System's Base.
  - Add the resulted value in step-B.

BINARY NUMBER SYSTEM TO DECIMAL

Binary Number is : **$(11000000111001)_2$**

1	1	0	0	0	0	0	0	1	1	1	0	0	1
13	12	11	10	9	8	7	6	5	4	3	2	1	0
$1x2^{13}$	$1x2^{12}$	$0x2^{11}$	$0x2^{10}$	$0x2^{9}$	$0x2^{8}$	$0x2^{7}$	$0x2^{6}$	$1x2^{5}$	$1x2^{4}$	$1x2^{3}$	$0x2^{2}$	$0x2^{1}$	$1x2^{0}$
8192	4096	0	0	0	0	0	0	32	16	8	0	0	1

=8192+4096+32+16+8+1

=12345

OCTAL NUMBER SYSTEM TO DECIMAL

Octal Number is : **$(30071)_8$**

3	0	0	7	1
4	3	2	1	0
$3*8^4$	$0*8^3$	$0*8^2$	$7*8^1$	$1*8^0$
12288	0	0	56	1

=12288+0+0+56+1

=12345

Decimal Number is: **$(12345)_{10}$**

HEXADECIMAL NUMBER SYSTEM TO DECIMAL

Hexadecimal Number is : $(2D5)_{16}$

2	D (13)	5
2	1	0
$2*16^2$	$13*16^1$	$5*16^0$
512	208	5

=512+208+5

=725

Decimal Number is: $(725)_{10}$

UNIT - 1

DATA REPRESENTATION

Dr. Saurabh Pal
Associate Professor & Head
Department of Computer Applications
Veer Bahadur Singh Purvanchal University, Jaunpur

OBJECTIVES

- Description of Coding Scheme
- ASCII
- EBCDIC
- Unicode

DESCRIPTION OF CODING SCHEME

- Coding schemes are a common way of representing a character of data. It is required in computers for exchanging data. The following are a few common coding schemes-
  - ASCII: It stands for the American Standard Code for Information Interchange. It is used on almost all computers, hence considered as a standard coding scheme.
  - EBCDIC: It stands for Extended Binary Coded Decimal Interchange Code. Its is primarily used in IBM and IBM-compatible mainframes.
  - Unicode: It is designed to accommodate alphabets (-256). It uses 16 bits to represent one character and requires twice as much space to store data. It can have a maximum of 65,536 possible values.

ASCII

- ASCII stands for American Standard Code for Information Interchange
- The code uses 7 bits to encode 128 unique characters
- As a note, formally, work to create this code began in 1960. 1st standard in 1963. Last updated in 1986.

ASCII EXAMPLE

- Encoding of 123
  - 011 0001 011 0010 011 0011
- Encoding of Joanne
  - 100 1010 110 1111 110 0001
  - 110 1110 110 1110 110 0101
- Note that these are 7 bit codes

ASCII EXAMPLE 8TH BIT

- In digital systems data is usually organized as bytes or 8 bit of data.
- How about using the 8th bit for an error coding. This would help during data transmission, etc.
- Parity bit – the extra bit included to make the total number of 1s in the byte either even or odd – called even parity and odd parity

EXAMPLE OF PARITY

- Consider data 100 0001
  - Even Parity 0100 0001
  - Odd Parity 1100 0001
- Consider data 1010100
  - Even Parity 1101 0100
  - Odd Parity 0101 0100
- A parity code can be used for ASCII characters and any binary data.

EBCDIC CODE

- The EBCDIC stands for Extended Binary Coded Decimal Interchange Code.
- IBM invented this code to extend the Binary Coded Decimal which existed at that time. All the IBM computers and peripherals use this code.
- It is an 8 bit code and therefore can accommodate 256 characters.

UNICODE

- Unicode is the newest concept in digital coding.
- In Unicode every number has a unique character.
- Leading technological giants have adopted this code for its uniqueness.