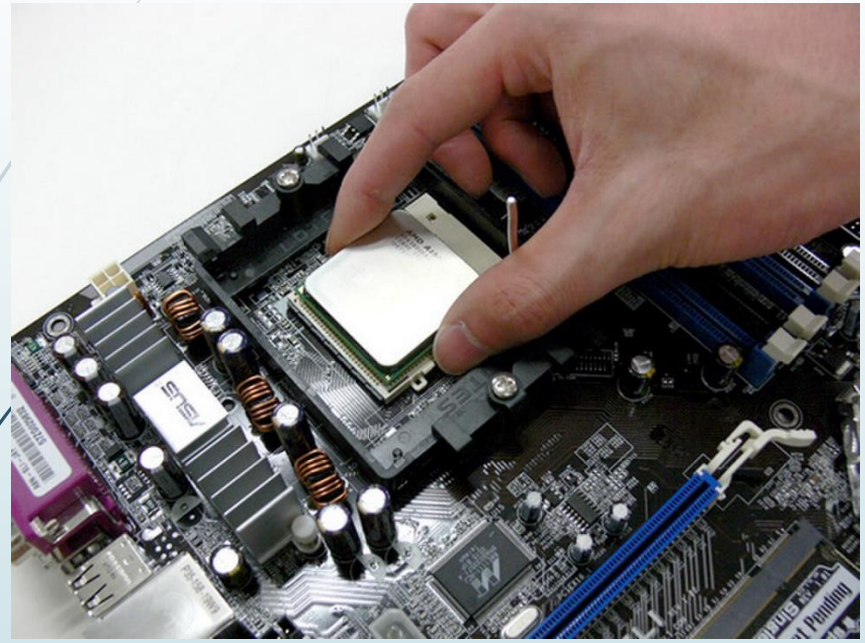
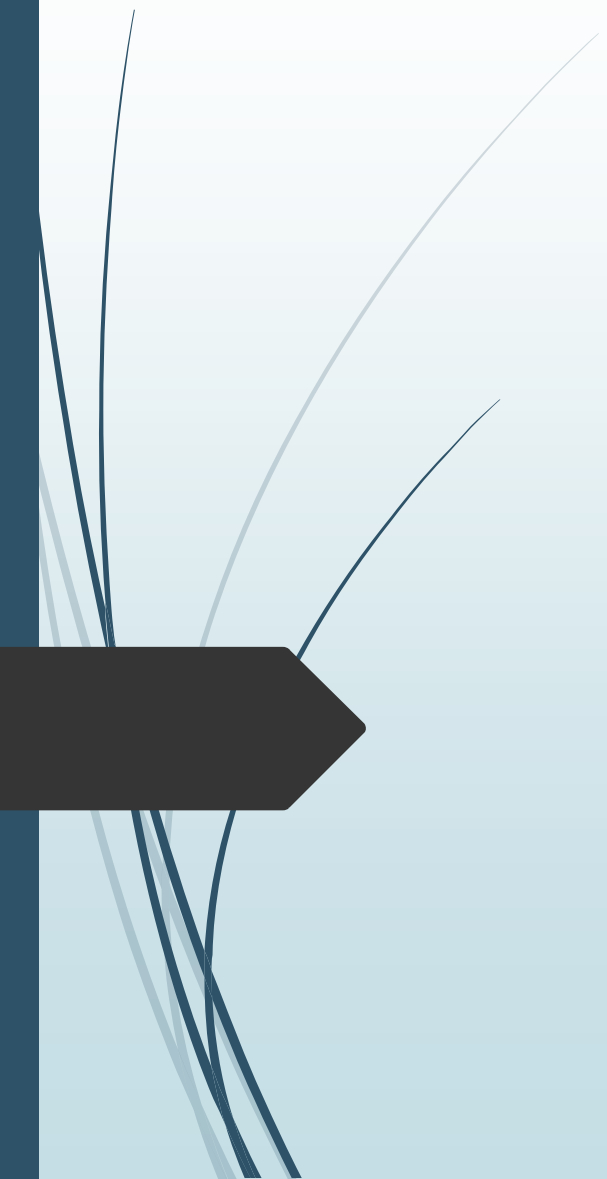
جامعة برج العرب
التكنولوجية



IT Essentials

د/ اسامه النحاس



Data Representation

Decimal System

- A numbering system is a way for representing certain value in different ways.

- Decimal system (base = 10): We have 10 symbols to represent values (0,1,2,3,4,5,6,7,8,9).
- When finishing all the symbols in a digit, we make it zero and add one to the next digit.

$Value_{10}$
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

Decimal System

- Decimal number system is base 10
  - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  - Uses 10 numbers

23625

Power of 10 representation	**10^4**	**10^3**	**10^2**	**10^1**	**10^0**
Decimal representation	**10000**	**1000**	**100**	**10**	**1**
Base 10 representation	**20,000**	**3,000**	**600**	**20**	**5**

Binary System

- Binary system (base = 2): We have 2 symbols to represent values (0,1).
- When finishing all the symbols in a digit (bit), we make it zero and add one to the next digit (bit).
- We need too many bits for representing relatively small values:

$$(1000)_{10} = (11\ 1110\ 1000)_2$$

$$(1{,}000{,}000)_{10} =$$

$$(1111\ 0100\ 0010\ 0100\ 0000)_2$$

$Value_2$	$Value_{10}$
0	0
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15
10000	16

Converting Binary to Decimal

Binary number system is base 2

- **0, 1**
- **Uses 2 numbers**

10010001 = 145

Base 2 representation	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal representation	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Base 2 representation	**1**	**0**	**0**	**1**	**0**	**0**	**0**	**1**

Converting to Decimal

- To find the value of any representation in any numbering system we multiply each digit by its corresponding weight.
- Binary system: Example value of $(1101)_2$:

Digits	1	1	0	1
Weights	2^3	2^2	2^1	2^0
Weights	8	4	2	1
Weighted digits	1×8	1×4	0×2	1×1
Value	$(13)_{10}$			

Convert Binary to Decimal

1. Choose an 8 bit binary number = 10101110
2. Write the binary digits under the correct column
3. For each column with a 1, you will add that decimal value
4. You will not add the values of the columns you entered 0

Power of 2 representation	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal representation	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Base 2 representation	**1**	**0**	**1**	**0**	**1**	**1**	**1**	**0**

128 + 32 + 8 + 4 + 2 = 174

10101110 = 174

Hexadecimal System

- Hexadecimal system (base = 16): We have 16 symbols to represent values (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F).
- When finishing all the symbols in a digit, we make it zero and add one to the next digit.
- Significantly less bits are required to represent the same values:

$$(1000)_{10} = (3E8)_{16}$$

$$(1,000,000)_{10} = (F4240)_{16}$$

$Value_2$	$Value_{10}$	$Value_{16}$
0	0	**0**
1	1	**1**
10	2	**2**
11	3	**3**
100	4	**4**
101	5	**5**
110	6	**6**
111	7	**7**
1000	8	**8**
1001	9	**9**
1010	10	**A**
1011	11	**B**
1100	12	**C**
1101	13	**D**
1110	14	**E**
1111	15	**F**
10000	16	**10**

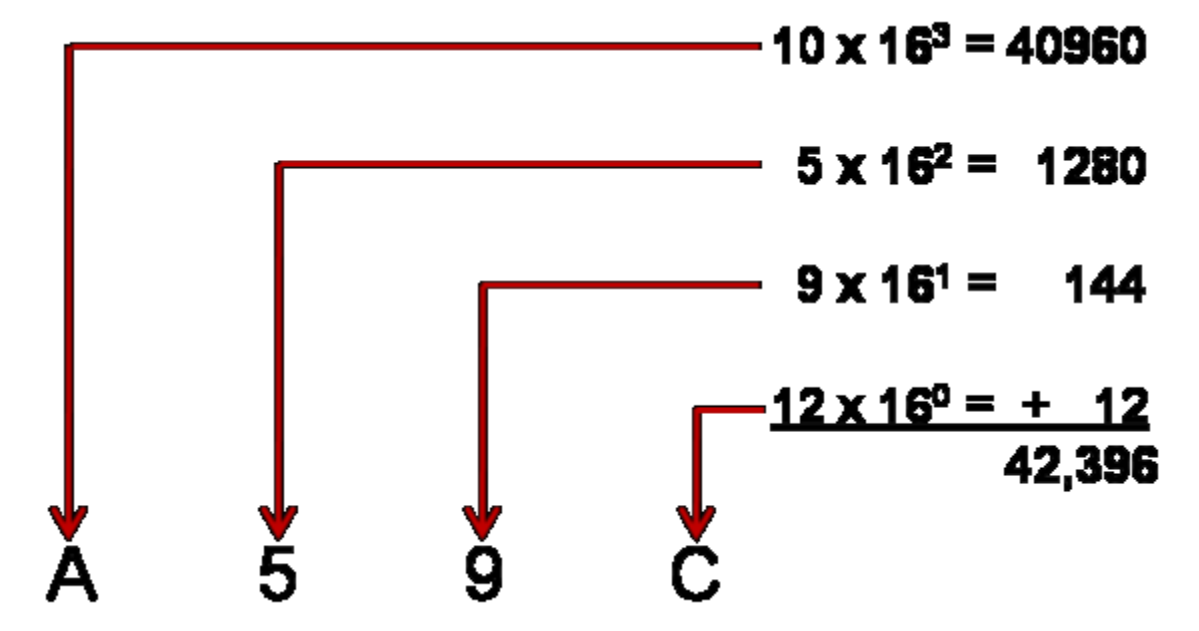
Converting to Decimal

- To find the value of any representation in any numbering system we multiply each digit by its corresponding weight.
- Hexadecimal system: Example value of $(2A)_{16}$:

Digits	2	A
Weights	16^1	16^0
Weights	16	1
Weighted digits	2×16	A×1=?
Value	$(42)_{10}$	

A	10
B	11
C	12
D	13
E	14
F	15

Hexa to decimal



Octal System

- Octal system (base = 8): We have 8 symbols to represent values (0,1,2,3,4,5,6,7).
- When finishing all the symbols in a digit, we make it zero and add one to the next digit.
- Number of representable values in N digits: $M = 8^N$
- Range of values: $0 \rightarrow M - 1$

$Value_2$	$Value_8$	$Value_{10}$	$Value_{16}$
0	**0**	0	0
1	**1**	1	1
10	**2**	2	2
11	**3**	3	3
100	**4**	4	4
101	**5**	5	5
110	**6**	6	6
111	**7**	7	7
1000	**10**	8	8
1001	**11**	9	9
1010	**12**	10	A
1011	**13**	11	B
1100	**14**	12	C
1101	**15**	13	D
1110	**16**	14	E
1111	**17**	15	F
10000	**20**	16	10

Converting to Decimal

- To find the value of any representation in any numbering system we multiply each digit by its corresponding weight.
- Octal system: Example value of $(263)_8$:

Digits	2	6	3
Weights	8^2	8^1	8^0
Weights	64	8	1
Weighted digits	2×64	6×8	3×1
Value		$(179)_{10}$	

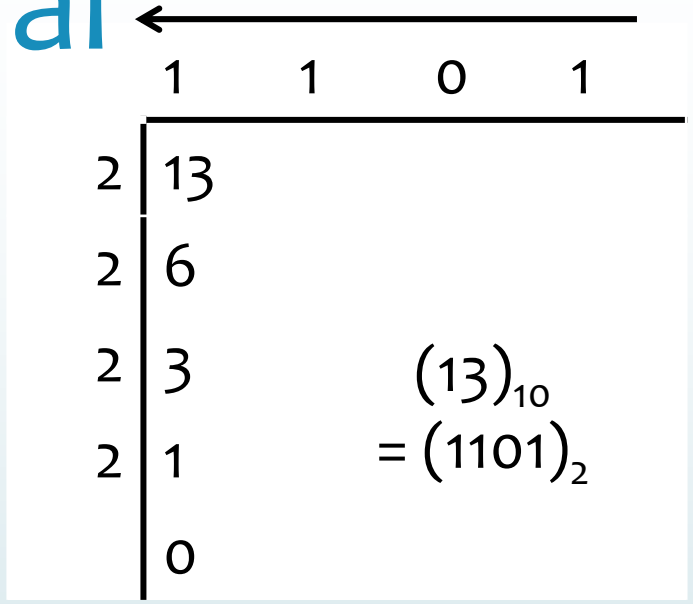
Value of Representation

- To find the value of any representation in any numbering system we multiply each digit by its corresponding weight.
- Decimal system: Example value of $(7129.45)_{10}$:

Digits	7	1	2	9	.	4	5
Weights	10^3	10^2	10^1	10^0		10^{-1}	10^{-2}
Weights	1000	100	10	1		0.1	0.01
Weighted	2×1000	7×100	2×10	9×1		4×0.1	5×0.01
Value	7129.45						

Converting from Decimal

- Converting a decimal **integer** to binary (base 2):
  - Example: Given $(13)_{10}$
  - By iterative division by 2 till reaching 0.
  - The representation is the reminder.
- Converting a decimal **fraction** to binary (base 2):
  - Example: Given $(0.375)_{10}$
  - By iterative multiplication by 2 till reaching 0. Representation is the integer part.



$0.375 \times 2 = 0.75$
$0.75 \times 2 = 1.5$
$0.5 \times 2 = 1.0$
0.0

$\mathbf{(0.375)_{10} = (0.011)_2}$

Converting from Decimal

- Shortcut method for decimal to binary conversion:
  - Put ones into the bits corresponding to composing weights.
  - Example: Convert the numbers 9, 21, 12.25 into binary.

Weights	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}
Weights	32	16	8	4	2	1	0.5	0.25

9 =	0	0	1	0	0	1	. 0	0
21 =	0	1	0	1	0	1	. 0	0
12.25 =	0	0	1	1	0	0	. 0	1

Converting from Decimal

- Shortcut method for decimal to binary conversion:
  - Put ones into the bits corresponding to composing weights.
  - Example: Convert the numbers 9, 21, 12.25 into binary.

$$(9)_{10} = (1001)_2$$

$$(21)_{10} = (10101)_2$$

$$(12.25)_{10} = (1100.01)_2$$

9 =	0	0	1	0	0	1	. 0	0
21 =	0	1	0	1	0	1	. 0	0
12.25 =	0	0	1	1	0	0	. 0	1

Converting Decimal to Binary

- Convert decimal 35 to binary
  1. Using 8 bits, find largest power of 2 that will "fit" into 35
  2. Place a 1 into that slot
  3. If the # doesn't fit, place a 0 into that slot

Power of 2 representation	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal representation	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Base 2 representation	0	0	1	0	0	0	1	1

35 = 00100011

➢ Decimal into Binary

Decimal number 225

Division	Quotient	Remainder
225 / 2	112	1 ← LSB
112 / 2	56	0
56 / 2	28	0
28 / 2	14	0
14 / 2	7	0
7 / 2	3	1
3 / 2	1	1
1 / 2	0	1

Binary number 1 1 1 0 0 0 0 1

Decimal number 77

Division	Quotient	Remainder
77 / 2	38	1 ← LSB
38 / 2	19	0
19 / 2	9	1
9 / 2	4	1
4 / 2	2	0
2 / 2	1	0
1 / 2	0	1
		0

Binary number 0 1 0 0 1 1 0 1

Data Representation

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[END OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	|
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

Data Representation

- How is a letter converted to binary form and back?

Step 1.
The user presses the capital letter **D** (shift+D key) on the keyboard.

Step 2.
An electronic signal for the capital letter **D** is sent to the system unit.

Step 3.
The signal for the capital letter **D** is converted to its ASCII binary code (01000100) and is stored in memory for processing.

Step 4.
After processing, the binary code for the capital letter **D** is converted to an image, and displayed on the output device.

Binary Arithmetic (Addition)

Two bits	sum	Carry
0 + 0	0	0 (No carry)
0 + 1	1	0 (No carry)
1 + 0	1	0 (No carry)
1 + 1	0	1 (carry)

Three bits	sum	Carry
0 + 0 + 0	0	0 (No carry)
0 + 0 + 1	1	0 (No carry)
0 + 1 + 0	1	0 (No carry)
0 + 1 + 1	0	1
1 + 0 + 0	1	0 (No carry)
1 + 0 + 1	0	1
1 + 1 + 0	0	1
1 + 1 + 1	1	1

Binary Arithmetic (Addition)

Addition of the binary numbers involves the following steps—

1. Start addition by adding the bits in unit column (the right-most column). Use the rules of binary addition.
2. The result of adding bits of a column is a sum with or without a carry.
3. Write the sum in the result of that column.
4. If a carry is present, the carry is carried-over to the addition of the next left column.
5. Repeat steps 2–4 for each column, i.e., the tens column, hundreds column and so on.

Examples

Binary Addition	Decimal Addition
1 0 + 0 1 Result 1 1	2 + 1 3
$11_2 = 3_{10}$	

Binary Addition	Decimal Addition
1 1 ← Carry 0 1 + 1 1 Result 1 0 0	1 + 3 4
$100_2 = 4_{10}$	

Addition of Binary Numbers

The addition of any two signed binary numbers is performed as follows

- Represent the positive number in binary form.
- Represent the negative number in 2's complement form.
- Add the bits of the two signed binary numbers.
- Ignore any carry out from the sign bit position.
- Please note that the negative output is automatically in the 2's complement form.

Example: add 5 and 10.

Binary Addition	Decimal Addition
0 0 0 0 0 1 0 1 0 0 0 0 1 0 1 0 0 0 0 0 1 1 1 1	+ 5 + 1 0 + 1 5
The result is 0000 1111_2 i.e, $+15_{10}$	