
CSCI 250: EVERYTHING IS A NUMBER

PART I

Ayman Hajja, PhD

NUMBERING SYSTEMS

- Inside a computer, everything is a number
- The numbering system in a computer is called binary (only 0s and 1s) — Also called base-2 (since it uses two digits)
- What is the numbering system that we (humans) use?

Decimal — or Base-10 (since we use 10 symbols: 0 to 9)

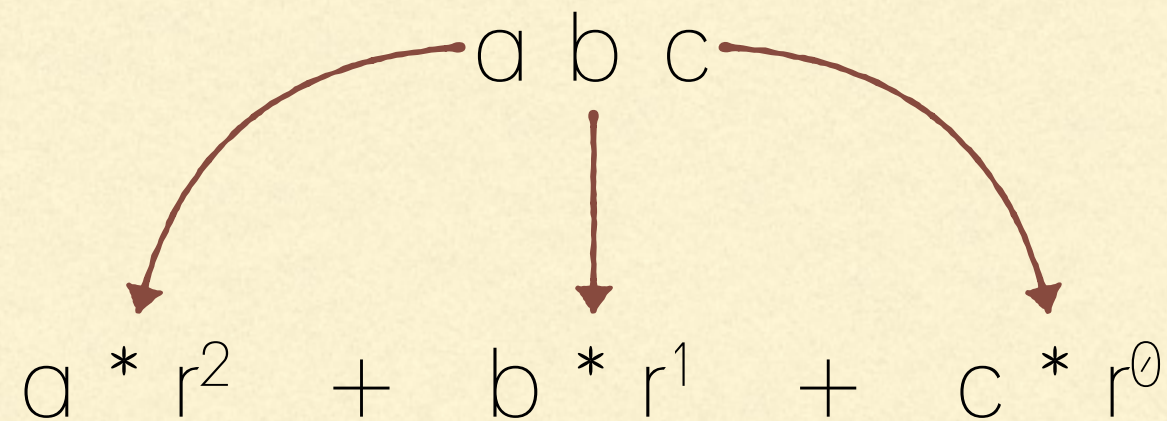
POSITIONAL VALUE/NOTATION (FOR DECIMALS)

- Positional notation (or place-value notation) is a method of representing or encoding numbers.
- The number 646 can be broken down to $600 + 40 + 6$

$$\begin{array}{c} 6 \quad 4 \quad 6 \\ \swarrow \quad \downarrow \quad \searrow \\ 6 * 10^2 + 4 * 10^1 + 6 * 10^0 \end{array}$$

POSITIONAL VALUE/NOTATION (FOR ANY OTHER BASE)

- Positional notation (or place-value notation) is a method of representing or encoding numbers.



... where r is the base of the numeral system

EXAMPLE 01

- What is the value of the number 421 given that the base is 5?

Can be written as 421_5 (subscript indicates base)

EXAMPLE 01

- What is the value of the number 421 given that the base is 5?

Can be written as 421_5 (subscript indicates base)

$$4 \cdot 5^2 + 2 \cdot 5^1 + 1 \cdot 5^0$$

Answer is 111_{10}

EXAMPLE 02

- What is the value of the number 121 given that the base is 4?

Can be rewritten as 121_4 (subscript indicates base)

EXAMPLE 02

- What is the value of the number 121 given that the base is 4?

Can be rewritten as 121_4 (subscript indicates base)

$$1 * 4^2 + 2 * 4^1 + 1 * 4^0$$

Answer is 25_{10}

TRUE FOR ANY BASE

- The number of digits is equal to the number of base:
 - Base-2: 0 and 1.
 - Base-10: 0 to 9
 - Zero is always one of the digits
 - The largest digit is the base minus one
 - The numeric value of a number represented by a sequence of digits is determined by the relative positions of the digits within the sequence
 - Larger numbers can be represented using less digits as the base increases
-

EXAMPLE 03

- What is the value of the number 101 given that the base is 2?

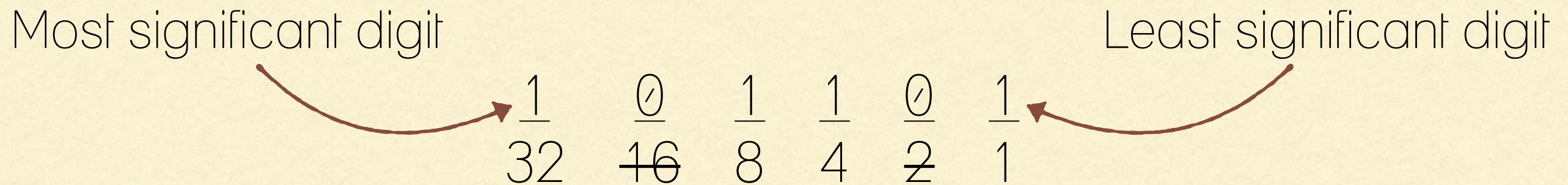
Can be written as 101_2 (subscript indicates base) — or $0b101$

$$1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$

Answer is 5_{10}

EASY WAY TO CONVERT FROM BINARY TO DECIMAL

- Starting with the least significant digit. List all the powers of two under each digit. Add up the values under digits 1.



The answer is $32 + 8 + 4 + 1 = 45$

BINARY NUMBER SYSTEM

- A binary digit is called a bit. 8 bits are called a byte
 - The least significant digit is commonly labeled LSD (similarly for the most significant digit — MSD)
 - The binary number system is used to model the series of electrical signals computers use to represent information. For example:
 - 0 might represent the no voltage, or ‘off’ state
 - 1 might represent the presence of voltage, or ‘on’ state
-

BINARY NUMBER SYSTEM

- How many unique combinations/numbers can be formed using three digits in the decimal system?

$10^3 = 1000$: From 000 to 999

- How many unique combinations/numbers can be formed using three digits in the binary system?

$$2^3 = 8:$$

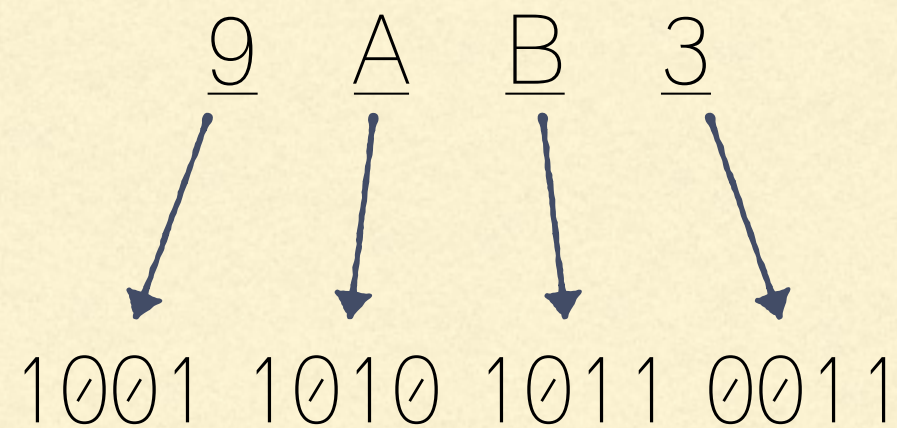
000 — 001 — 010 — 011 — 100 — 101 — 110 — 111

BASE 16 (HEXADECIMAL)

- Uses 16 distinct symbols: from 0 to 9, and A, B, C, D, E, F to represent values ten to fifteen respectively
 - Can be easily converted to binary (and vice versa):
 - Each hexadecimal digit represents four binary digits
-

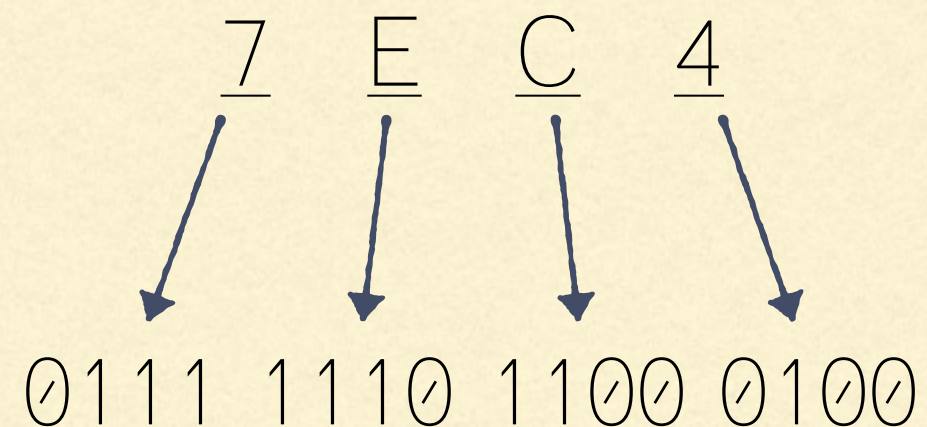
CONVERTING FROM HEXADECIMAL TO BINARY

- To convert from hexadecimal to binary, we substitute each hexadecimal digit with its corresponding 4 bits (binary digits)



CONVERTING FROM HEXADECIMAL TO BINARY — ONE MORE EXAMPLE

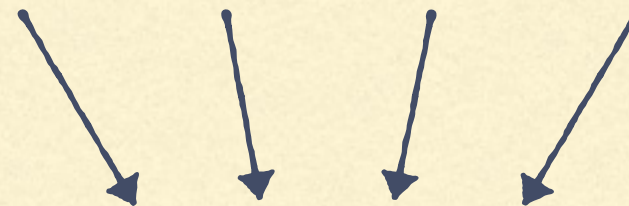
- Convert the following hexadecimal to binary:



CONVERTING FROM BINARY TO HEXADECIMAL

- To convert from binary to hexadecimal, we substitute each four binary bits with its corresponding hexadecimal digit

0111 1110 1100 0100



7 E C 4

CONVERTING FROM DECIMAL TO BINARY

- Any ideas?

CONVERTING FROM DECIMAL TO BINARY

1. List the powers of two in a "base 2 table" from right to left. Start at 2^0 , evaluating it as "1"

512	256	128	64	32	16	8	4	2	1

2. Choose the biggest number x that will fit into the number you are converting
 3. Write a 1 above this box in your chart for the leftmost binary digit. Then, subtract x from your initial number, and repeat
-

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

512	256	128	64	32	16	8	4	2	1

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

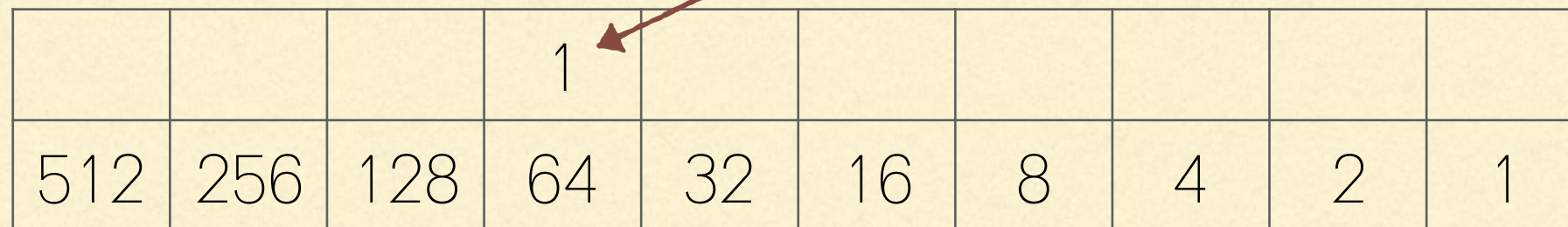
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

We place 1 here



			1						
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

			1						
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

$$93 - 64 = 29$$

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

			1						
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

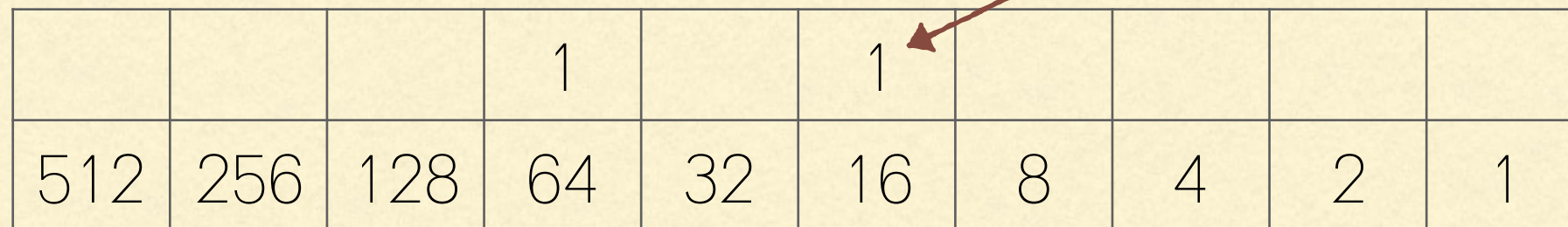
$$93 - 64 = 29$$

16 is the largest number that is a power of two and less than 29

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

We place 1 here



			1		1				
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

$$93 - 64 = 29$$

16 is the largest number that is a power of two and less than 29

CONVERTING FROM DECIMAL TO BINARY — EXAMPLE 01

- Convert the decimal number 93 to binary

			1	0	1	1	1	0	1
512	256	128	64	32	16	8	4	2	1

64 is the largest number that is power of two and less than 93

$$93 - 64 = 29$$

16 is the largest number that is a power of two and less than 29

Repeat...

MORE EXERCISES

- Convert the following numbers from their initial base/radix to the other two common bases/radices:

A. 0b10010011

B. 0xAD

C. 0x7E

D. 63

E. 0b100100

BINARY ADDITION

- Four possible binary addition combinations:

$$\begin{array}{r} (1) \quad 0 \\ + 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} (2) \quad 1 \\ + 0 \\ \hline 1 \end{array}$$

$$\begin{array}{r} (3) \quad 0 \\ + 1 \\ \hline 1 \end{array}$$

$$\begin{array}{r} (4) \quad 1 \\ + 1 \\ \hline 10 \end{array}$$

ADDING BINARY — OVERFLOW

- More complicated binary additions are also done just like we do decimal addition. Let's do an 8-bit addition:

$$\begin{array}{r} 00001110 \\ + 00000101 \\ \hline \end{array}$$

00010011

No overflow

$$\begin{array}{r} 10001110 \\ + 10001101 \\ \hline \end{array}$$

00011011

Overflow Problem

REPRESENTING NEGATIVE VALUES.

WAY 1: SIGN & MAGNITUDE

- Use one bit (usually the left-most/most-significant) to indicate the sign.
 - "0" indicates a positive integer,
 - "1" indicates a negative integer.
- Question: With 8-bit sign-magnitude representation, what positive integers can be represented and what negative integers can be represented?

REPRESENTING NEGATIVE VALUES.

WAY 1: SIGN & MAGNITUDE

- Use one bit (usually the left-most/most-significant) to indicate the sign.
 - "0" indicates a positive integer,
 - "1" indicates a negative integer.
- Question: With 8-bit sign-magnitude representation, what positive integers can be represented and what negative integers can be represented?

-127 ... 0 ... 127

WAY 1: SIGN & MAGNITUDE

- There are several problems with sign-magnitude. It works well for representing positive and negative integers (although the two zeros are bothersome). But it does not work well in computation. A good representation method (for integers or for anything) must not only be able to represent the objects of interest, but must also support operations on those objects.
- (4 bit binary) Can the "binary addition algorithm" be used with sign-magnitude representation? Try adding +7 with -4?

The answer is no
