Introduction to Computer Science (COS 101)

Week I

Dr. Desmond Moru
dmoru@pau.edu.ng

Chudi Ofoma
cofoma@pau.edu.ng

SCHOOL OF SCIENCE AND TECHNOLOGY
PAN-ATLANTIC UNIVERSITY

What is Computer Science?

Computer Science can be defined as the study of computers and computational systems.

Computer scientists deal mostly with software and software systems; this includes their theory, design, development, and application

It deals with:

- The theory of computation
- Algorithms,
- Computational problems and
- The design of computer systems; hardware, software and applications.

What is Computer Science?

Computer Science addresses both human-made and natural information processes, such as:

- Communication
- Control
- Perception
- Learning and
- Intelligence, especially in computing systems and machines.

Characteristics of a Computer

SPEED:	• In general, no human being or any other device can compete to solve complex computation, faster than computer.
ACCURACY:	• It does computation with high precision. In other words, it gives result with high degree of accuracy
STORAGE:	• Computer has the capacity to store mass volume of data with appropriate format.
DILIGENCE:	• Computer never feels bored and it can work for hours and hours without any break and creating error.
VERSATILITY:	• Computers are multipurpose. A Computer can perform completely different types of work at the same time.
POWER OF REMEMBERING:	• It stores data for future use.
NO IQ:	• Computer needs to be instructed to do its job.
NO FEELING:	• Computer does not have emotions, knowledge, experience and feeling.

The Universal Machine

- A modern computer can be defined as a machine that stores and manipulates information under the control of a changeable program.
- It is an electronic device that accepts data, performs computations, and makes logical decision according to instructions that have been given to it; then produces meaningful information in a form that is useful to humans.

What is a Computer Program?

- A computer program is a detailed, step-by-step set of instructions telling a computer exactly what to do.
- A computer program is a sequence of instructions written using a Computer Programming Language to perform a specified task by the computer.
- An unambiguous, ordered sequence of computational instructions necessary to achieve a solution.
- A computer program is a detailed plan or procedure for solving a problem.

What is Computer Science?

Its fields can be divided into theoretical and practical disciplines. E.g.:

- Computational complexity theory
- Computer graphics and computational geometry
- Algorithms
- Programming language
- Software engineering
- Computer architecture and computer engineering
- Human–Computer Interaction (HCI)
- Artificial intelligence

Historical Perspectives

First generation	• 1946 – 1959
Second generation	• 1959 - 1965
Third generation	• 1965 – 1971
Fourth generation	• 1971 -1980
Fifth generation	• 1980 – till date

Characteristics of First Generation Computers

Main electronic component	• Vacuum tube
Programming language	• Machine Language
Main Memory	• Magnetic tapes and magnetic drums.
Input/output devices	• Paper tape and punched cards.
Speed and size	• Very slow and very large in size (often taking up entire room)
Examples of the first generation	• IBM 650, IBM 701, ENIAC, UNIVAC1, etc.

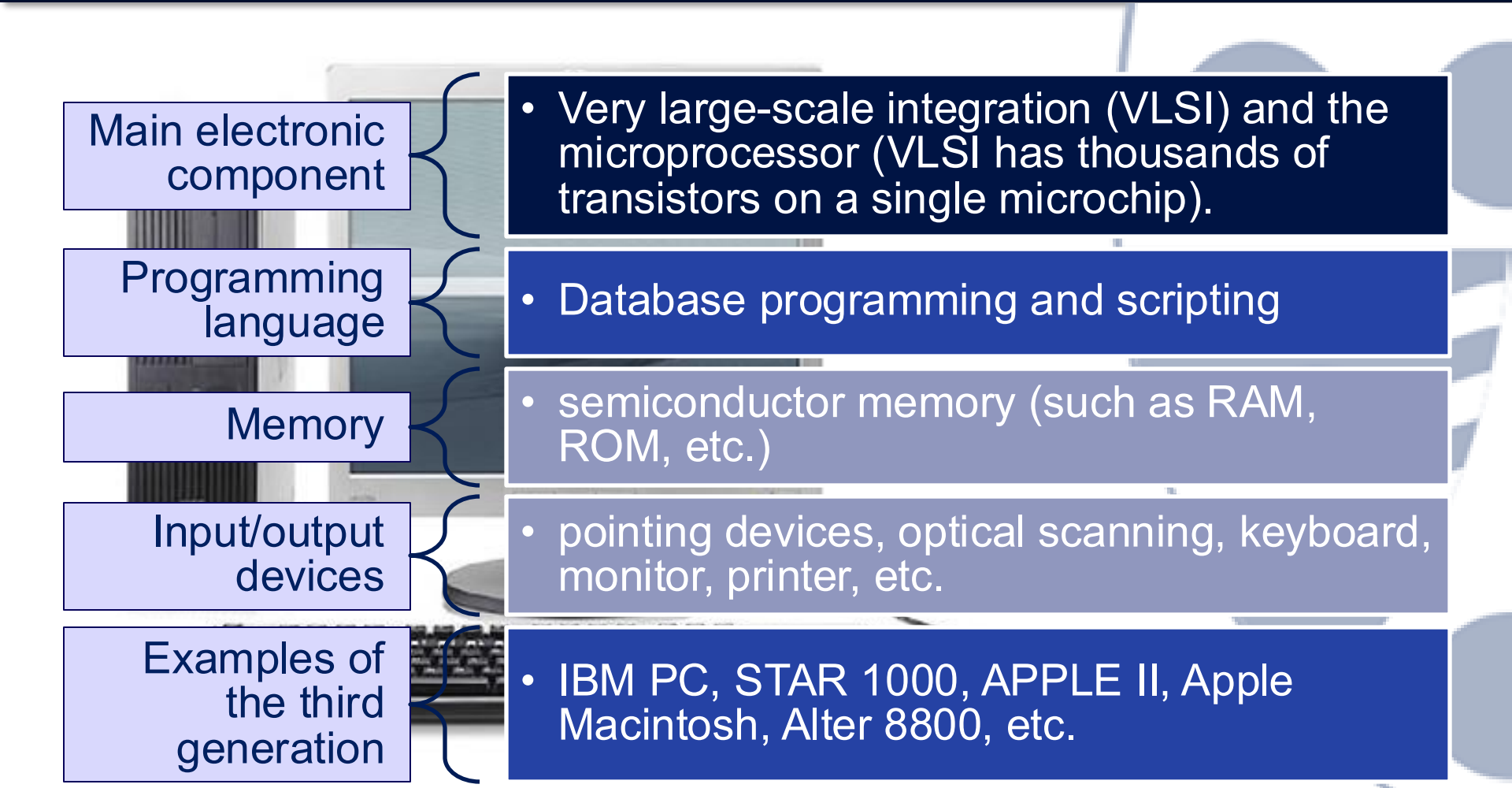
Characteristics of Second Generation Computers

Main electronic component	• Transistor
Programming language	• Low-level and assembly language
Memory	• Magnetic core and magnetic tape/disk.
Input/output devices	• Magnetic tape and punched cards.
Power and size	• Smaller in size, low power consumption, and generated less heat (in comparison with the first generation computers).
Examples of the second generation	• PDP-8, IBM1400 series, IBM 7090 and 7094, UNIVAC 1107, CDC 3600 etc.

Characteristics of Third Generation Computers

Main electronic component	• Integrated circuits (ICs)
Programming language	• High-level language
Memory	• Large magnetic core, magnetic tape/disk
Input/output devices	• Magnetic tape, monitor, keyboard, printer, etc.
Examples of the third generation	• IBM 360, IBM 370, PDP-11, NCR 395, B6500, UNIVAC 1108, etc.

Characteristics of Fourth Generation Computers



Main electronic component	• Very large-scale integration (VLSI) and the microprocessor (VLSI has thousands of transistors on a single microchip).
Programming language	• Database programming and scripting
Memory	• semiconductor memory (such as RAM, ROM, etc.)
Input/output devices	• pointing devices, optical scanning, keyboard, monitor, printer, etc.
Examples of the third generation	• IBM PC, STAR 1000, APPLE II, Apple Macintosh, Alter 8800, etc.

Characteristics of Fifth Generation Computers

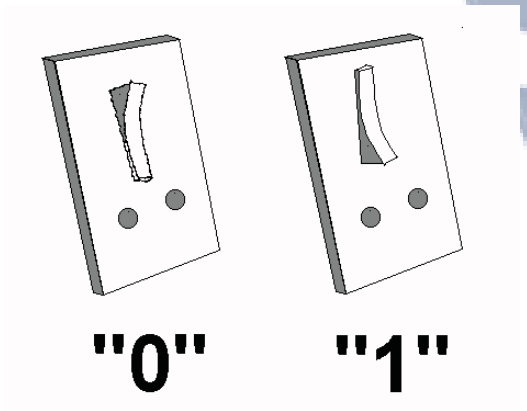
Main electronic component	• Based on artificial intelligence, uses the Ultra Large-Scale Integration (ULSI) technology and parallel processing method (ULSI has millions of transistors on a single microchip and Parallel processing method use two or more microprocessors to run tasks simultaneously).
Programming language	• Understand natural language (human language).
Size	• Portable and small in size.
Input/output devices	• Trackpad (or touchpad), touchscreen, pen, speech input (recognize voice/speech), light scanner, printer, keyboard, monitor, mouse, etc.
Examples of the third generation	• Desktops, laptops, tablets, smartphones, etc.

Bits and Data Storage

- Binary Numbers are the flow of information in the form of zeros and ones used by digital computers and systems.
- **BI**nary digi**TS** referred to as binary **BITS**

0 ⇨ OFF

1 ⇨ ON



Boolean Operation

- Bit 0 represents the value false and bit 1 represents the value true.
- Operations that manipulate true/false values are called Boolean operations, in honor of the mathematician George Boole (1815–1864), who was a pioneer in the field of mathematics called logic.
- Three of the basic Boolean operations are AND, OR, and XOR (exclusive or).

Boolean operations AND, OR, and XOR

The AND operation

0	0	1	1
AND 0	AND 1	AND 0	AND 1
0	0	0	1

The OR operation

0	0	1	1
OR 0	OR 1	OR 0	OR 1
0	1	1	1

The XOR operation

0	0	1	1
XOR 0	XOR 1	XOR 0	XOR 1
0	1	1	0

Gates and Flip-Flops

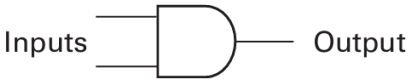
- A device that produces the output of a Boolean operation when given the operation's input values is called a gate.
- Inside today's computers, gates are usually implemented as small electronic circuits in which the digits 0 and 1 are represented as voltage levels.
- Gates provide the building blocks from which computers are constructed.

Gates and Flip-Flops

- A flip-flop is a fundamental unit of computer memory.
- It is a circuit that produces an output value of 0 or 1, which remains constant until a pulse (a temporary change to a 1 that returns to 0) from another circuit causes it to shift to the other value.
- The flip-flop is only one of many circuits that are basic tools in computer engineering.

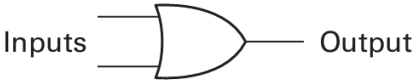
AND, OR, XOR and NOT Gate

AND



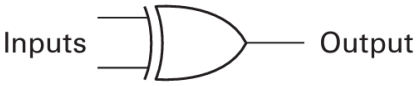
Inputs	Output
0 0 0 1 1 0 1 1	0 0 0 1

OR



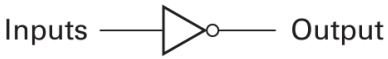
Inputs	Output
0 0 0 1 1 0 1 1	0 1 1 1

XOR



Inputs	Output
0 0 0 1 1 0 1 1	0 1 1 0

NOT



Inputs	Output
0 1	1 0

Binary to Decimal Conversion

- Conversion of binary to decimal (base-2 to base-10) numbers and back is an important concept to understand as the binary numbering system forms the basis for all computer and digital systems.

The Decimal Numbering System

- Each number column has values of units, tens, hundreds, thousands, etc.
- Mathematically these values are written as 10^0, 10^1, 10^2, 10^3, etc.
- Each position to the left of the decimal point indicates an increased positive power of 10.

The Decimal Numbering System

- The value of any decimal number is equal to the sum of its digits multiplied by their respective weights.
- For example: $N = 6163_{10}$
  - » $(6x1000) + (1x100) + (6x10) + (3x1) = 6163$

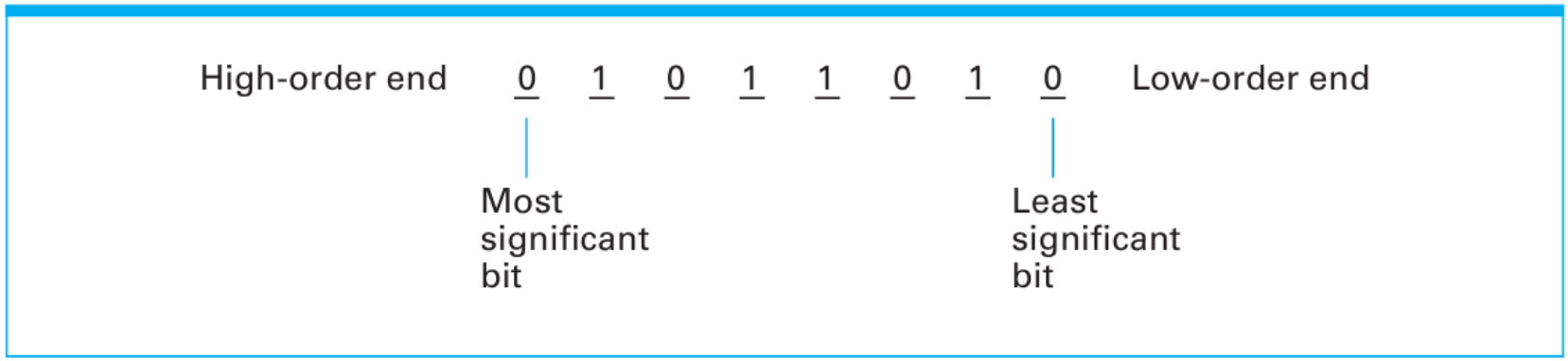
  OR

  - » $(6x10^3) + (1x10^2) + (6x10^1) + (3x10^0) = 6163$

The Binary Numbering System

- The most fundamental numbering system in all digital and computer based systems.
- Same set of rules as the decimal numbering system.
- Unlike the decimal system which uses powers of ten, the binary number system works on powers of two.
    - » Base-2 to Base-10

Representation of a Binary Number



Most Significant Bit

Least Significant Bit

Convert a Binary to Decimal number

For example:

Decimal Digit Value	256	128	64	32	16	8	4	2	1
Binary Digit Value	1	0	1	1	0	0	1	0	1

By adding together all the decimal number values from right to left at the positions that are represented by a “1” gives us:

$(256) + (64) + (32) + (4) + (1) = 357_{10}$

Binary Number Names & Prefixes

Number of Binary Digits (bits)	Common Name
1	Bit
4	Nibble
8	Byte
16	Word
32	Double Word
64	Quad Word

Number of bytes

- 1000 bytes = 1 kilobyte (KB) (same as 1000^1 bytes)
- 1000 KB = 1 megabyte (MB) (same as 1000^2 bytes)
- 1000 MB = 1 gigabyte (GB) (same as 1000^3 bytes)
- 1000 GB = 1 terabyte (TB) (same as 1000^4 bytes)
- 1000 TB = 1 petabyte (PB) (same as 1000^5 bytes)
- 1000 PB = 1 exabyte (EB) (same as 1000^6 bytes)
- 1000 EB = 1 zettabyte (ZB) (same as 1000^7 bytes)
- 1000 ZB = 1 yottabyte (YB) (same as 1000^8 bytes) I will not demand for more than this level
- 1000 YB = 1 xenottabyte (XB)
- 1000 XB = 1 shilentnobyte (SB)
- 1000 SB = 1 domegemegrottebyte (DB)

Hexadecimal Numbers

- Hexadecimal Numbers group binary numbers into sets of four allowing for the conversion of 16 different binary digits.
- The “Hex” numbering system uses the Base of 16 system.
- The one main disadvantage of binary numbers is that the binary string equivalent of a large decimal base-10 number can be quite long.

Hexadecimal Numbers

Decimal number	4-bit Binary Number	Hexadecimal Number
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hexadecimal Numbers Example

Convert the following Binary number $1110\ 1010_2$ into its Hexadecimal number equivalent.

11101010_2			
Group the bits into four's starting from the right hand side			
=	1110	1010	
Find the Decimal equivalent of each individual group			
=	14	10	(in decimal)
Convert to Hexadecimal using the table above			
=	E	A	(in Hex)
Then, the hexadecimal equivalent of the binary number $1110\ 1010_2$ is $\#EA_{16}$			

Coding Systems for Text-Based Data

- While numeric data is represented by the binary numbering system, text-based data is represented by binary coding systems specifically developed for text-based data — namely, ASCII, EBCDIC, and Unicode.
- These codes are used to represent all characters that can appear in text data—such as numbers, letters, and special characters and symbols like the dollar sign, comma, percent symbol, and mathematical symbols.

ASCII and EBCDIC

- ASCII (American Standard Code for Information Interchange) is the coding system traditionally used with personal computers.
- EBCDIC(Extended Binary-Coded Decimal Interchange Code) was developed by IBM, primarily for use with mainframes.

Unicode

- Unlike ASCII and EBCDIC, which are limited to only the Latin alphabet used with the English language, Unicode is a universal international coding standard designed to represent text-based data written in any ancient or modern language, including those with different alphabets, such as Chinese, Greek, Hebrew, Amharic, Tibetan, and Russian.