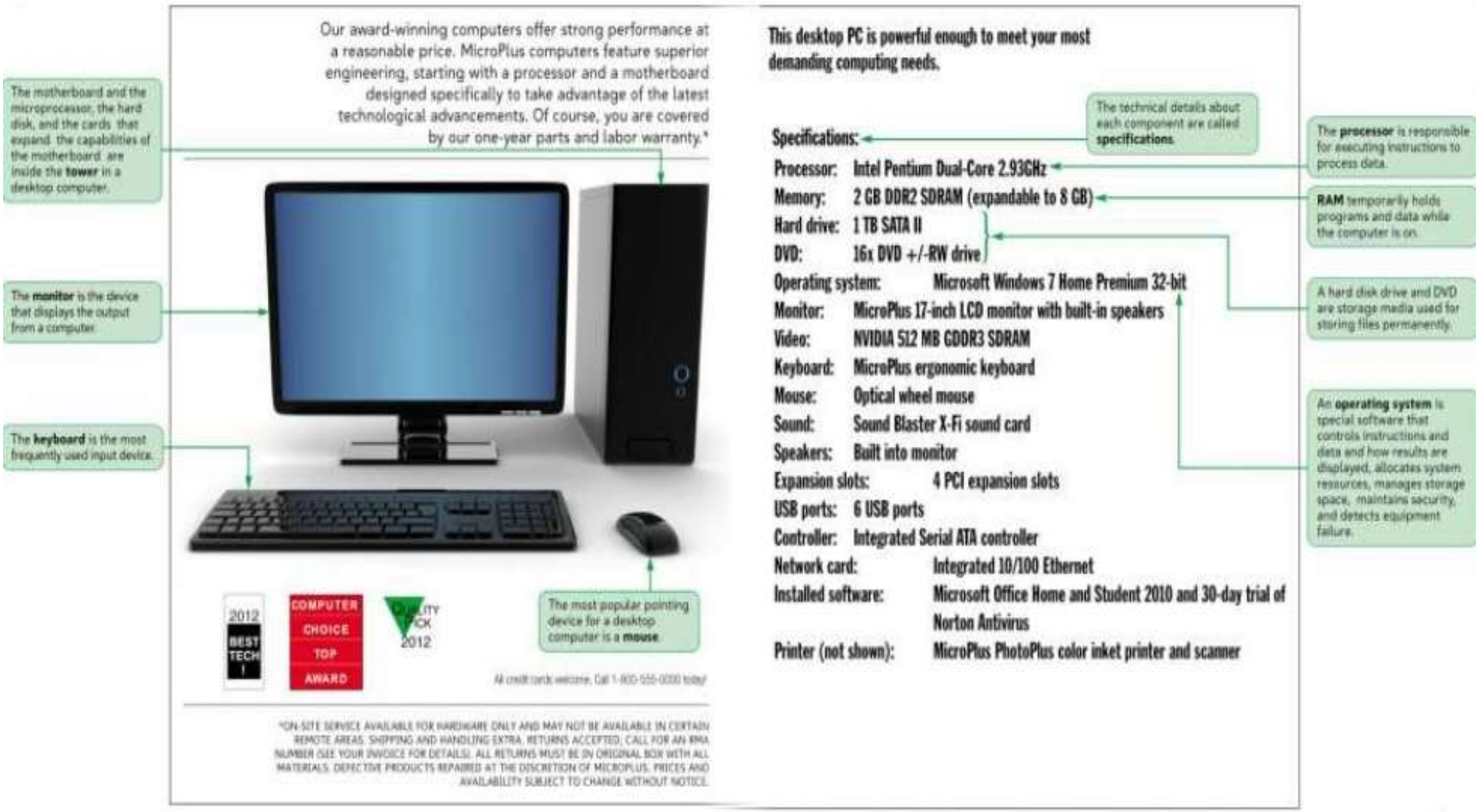
UNIT 3

Computer System

Nguyen Thi Thu Huong-SoICT-HUST

An advertisement of a computer



Unit 3. Computer Systems

- Computer Architecture
- Computer Software

Nguyen Thi Thu Huong-SoICT-HUST

Computer Architecture

- General Model of a Computer
- The Central Processing Unit (CPU)
- Memory
- Input-Output Devices
- Buses

General Model of a Computer

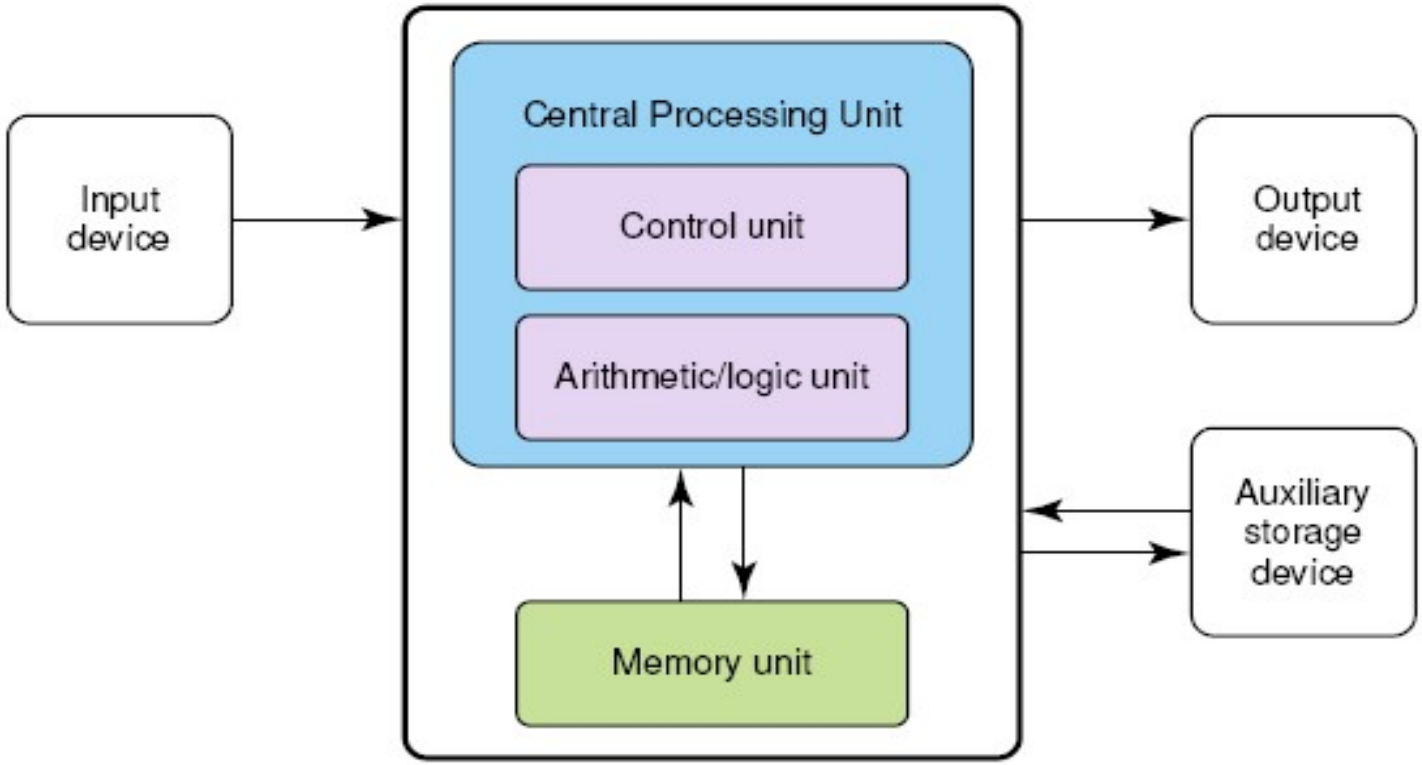
- Computer Operations
- Architecture of Computer Systems
- The Fetch – Execute Cycle

Computer Operations

A computer is a programmable electronic device that can store, retrieve and process data

- Data Processing
- Data Storage
- Data Retrieve
- Programmable

The von Neuman Architecture



Parts that Build Up A System Unit

- Casing or cover
- Power Supply
- Motherboard
- Microprocessor
- Memory
- Video Card
- Sound card
- Floppy disk drive
- Hard disk drive
- CD-ROM drive
- MODEM

CASING OR COVER

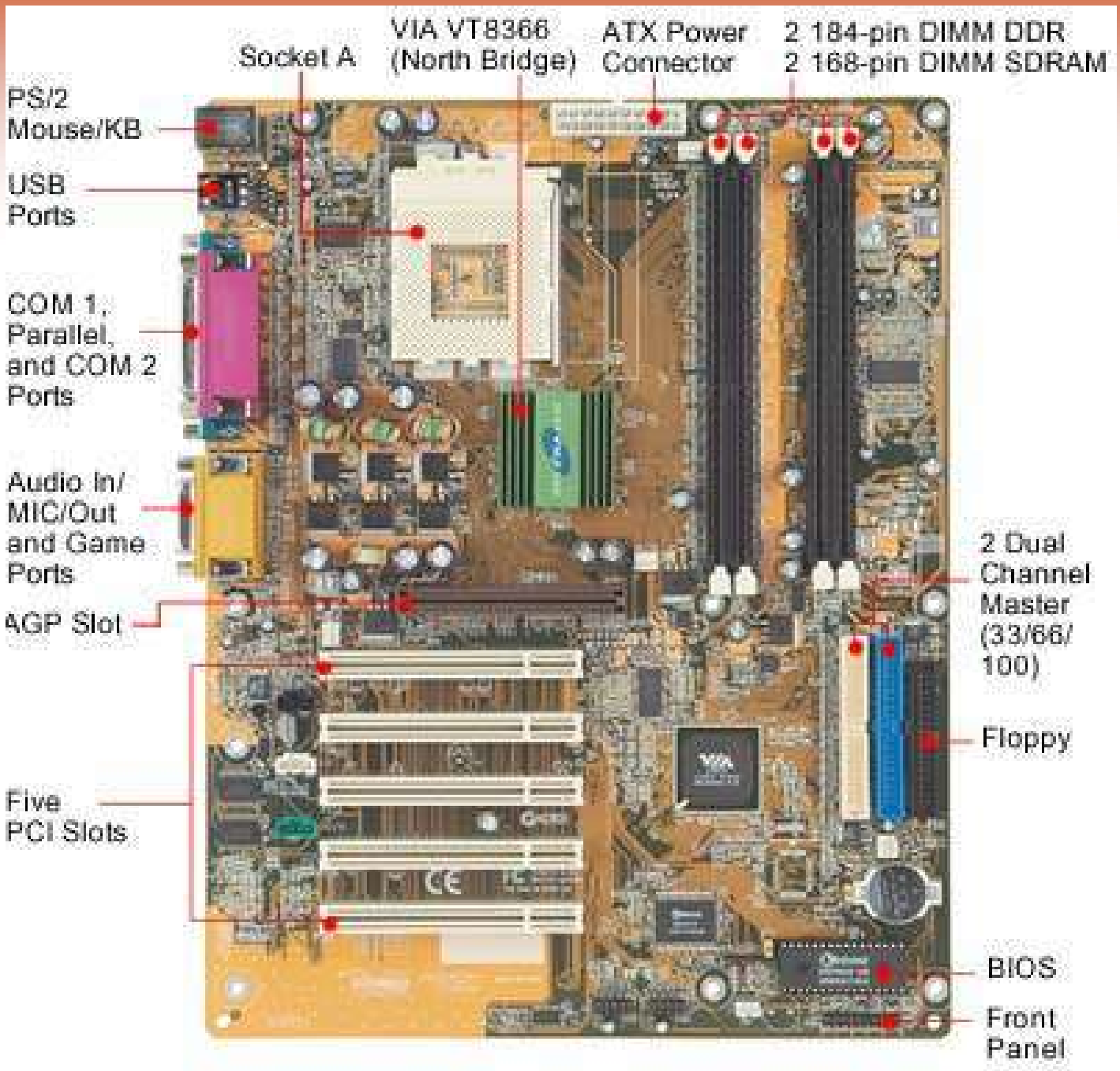
- The box or outer shell that houses most of the computer, it is usually one of the most overlooked parts of the PC.
- Protects the computer circuits, cooling and system organization.

Power Supply

- Responsible for powering every device in your computer.
- Parts of a Power supply:
  - Disk drive connectors
  - Motherboard connector
  - Power supply fan
  - Power switch
  - Input voltage selector
  - Cover
  - Power plugs receptacle

Motherboard

- The physical arrangement in a computer that contains the computer's basic circuitry and components.
- Components are:
  - Microprocessor
  - (Optional) Coprocessors
  - Memory
  - Basic Input/Output System (BIOS)
  - Expansion Slot
  - Interconnecting circuitry


Socket A
VIA VT8366
(North Bridge)
ATX Power
Connector
2 184-pin DIMM DDR
2 168-pin DIMM SDRAM
PS/2
Mouse/KB
USB
Ports
COM 1,
Parallel,
and COM 2
Ports
Audio In/
MIC/Out
and Game
Ports
AGP Slot
Five
PCI Slots
2 Dual
Channel
Master
(33/66/
100)
Floppy
BIOS
Front
Panel

Expansion Slots

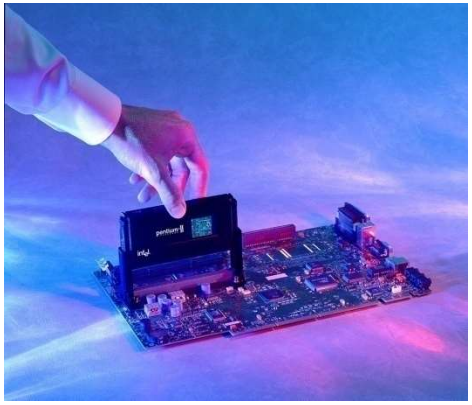
- Graphic cards
- Sound cards
- Modem cards
- Network interface cards/network adapter

Computer Organization

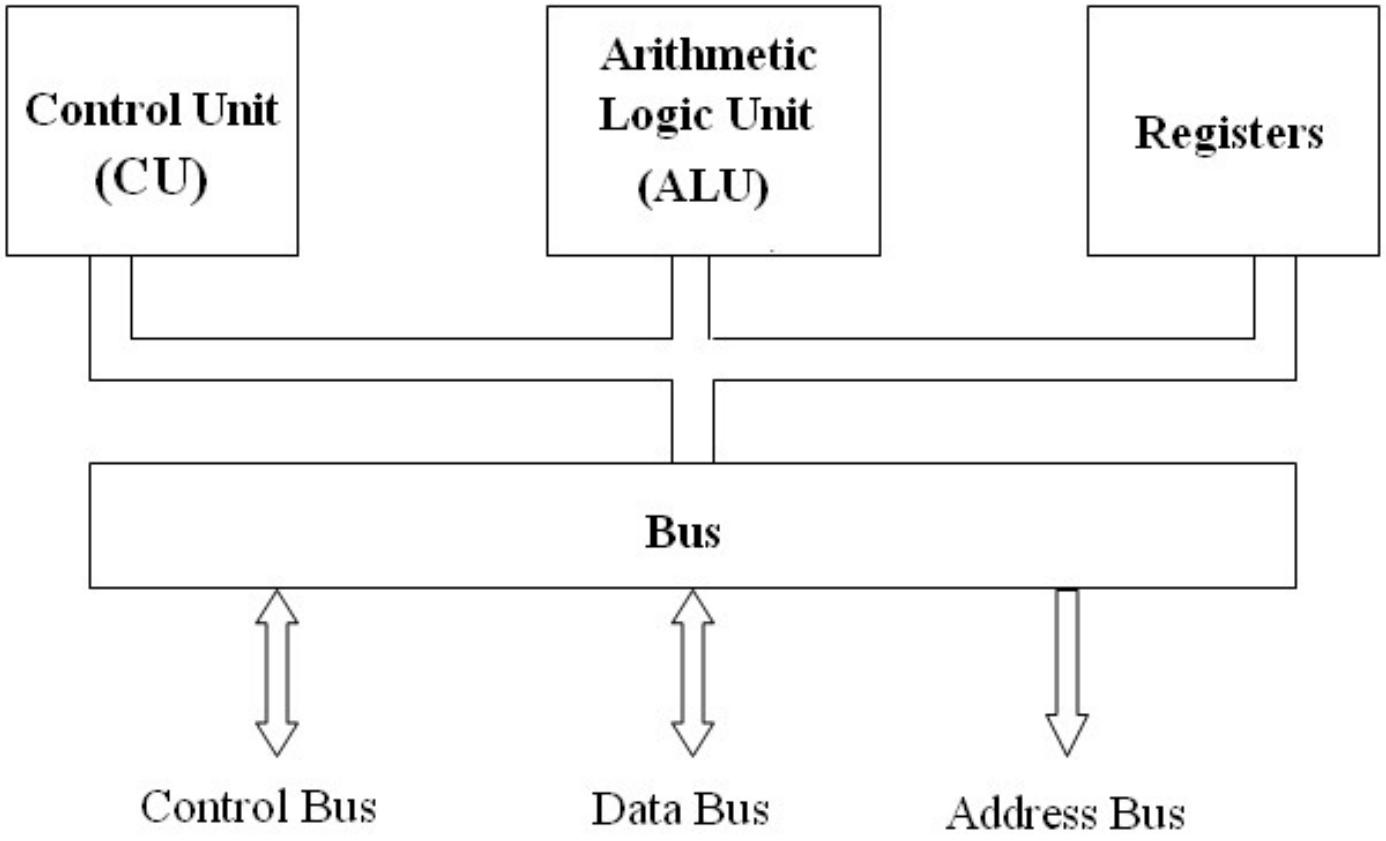
- CPU - central processing unit
  - Where decisions are made, computations are performed, and input/output requests are delegated
- Memory
  - Stores information being processed by the CPU: data and instructions
- Input devices
  - Moves data from outside word into the computer
- Output devices
  - Moves results from inside the computer to the outside world

THE CENTRAL PROCESSING UNIT (CPU)

- *Brains* of the computer
  - Arithmetic calculations are performed using the Arithmetic/Logical Unit or ALU
  - Control unit decodes and executes instructions
- Arithmetic operations are performed using binary number system
- CPU is contained on one (or a small number of) integrated circuits called microprocessors



Basic Model of the Central Processing Unit (CPU)



Arithmetic Logic Units (ALU)

- Performs the arithmetic and logical operations on the binary data .
- Most model ALU have small amount of registers

CONTROL UNIT

- In charge of the fetch-execute cycle
- Include 2 registers
  - IR (instruction register) : instruction to be executed
  - PC(program counter): the address of the next instruction

REGISTERS

- Temporary storage units within the CPU.
- Some registers, such as the program counter and instruction register, have dedicated uses.
- Other registers, such as the accumulator, are for more general purpose use.

Fetch-Execute Cycle

- Fetch the next instruction
- Decode the instruction
- Get data if needed
- Execute the instruction

CLOCK

- A circuit in a processor that generates a regular sequence of electronic pulses used to synchronize operations of the processor's components.
- Clock rate (frequency):The number of pulses per second
- The higher clock rate, the quicker speed of instruction processing

Memory

- Computer components, devices and recording media that retain digital data used for computing
- Includes internal and external memory

INTERNAL MEMORY

Accessible by a processor without the use of the computer input-output channels

- Main storage: RAM & ROM
- Cache memory
- Special registers

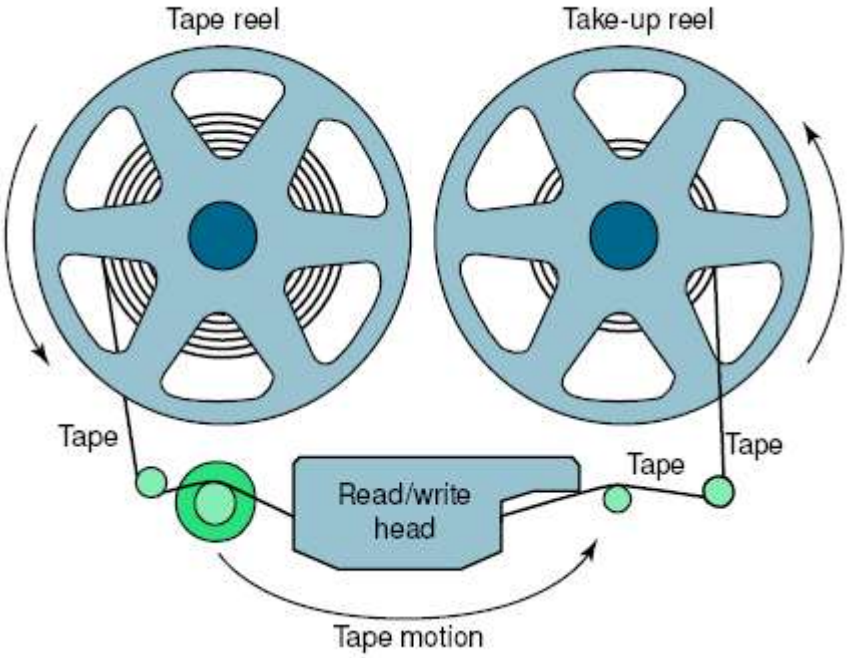
RAM AND ROM

- RAM (Random Access Memory), computer memory that can be read from and written to in arbitrary sequence.
- ROM Read-Only Memory, a class of storage media used in computers and other electronic devices. This tells the computer how to load the operating system.
- Storage capacity: the total amount of stored information that a storage device or medium can hold. It is expressed as a quantity of bits or bytes

External Memory (secondary storage)

- Holds information too large for storage in main memory.
- Information on external memory can only be accessed by the CPU if it is first t
- Ex rna r emory
- It etains i for na to keep a p rma

Floppy disk



pacity.
off, used

USB Flash Drive

Secondary Storage Devices

- Attached to the computer system to allow you to store programs and data permanently for the purpose of retrieving them for future use.
- Floppy disk, Hard disk, CD Rom

Floppy Disk

- The most common secondary storage device
- 3.5” disk – 1.44MB

High-Capacity Floppy Disks

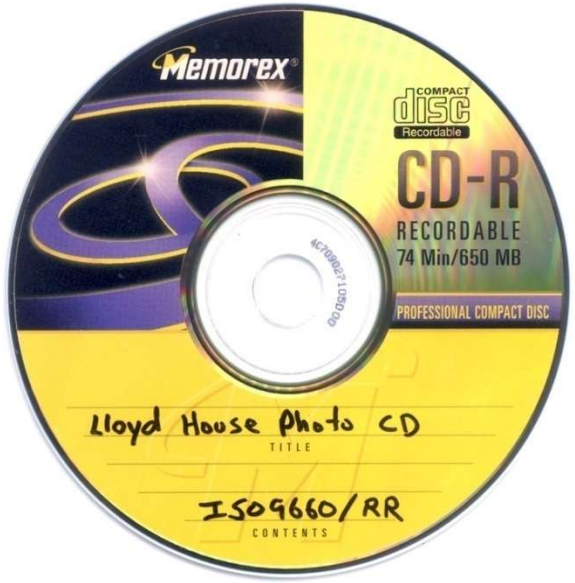
- Floppy disk cartridges
- 3 ½ inches in diameter
- Stores more information
- Zip disks

Hard Disk Drive or Hard Disk

- Made of rigid materials unlike floppy disks
- Holds a greater amount of data

Optical Discs

- A standard part of modern desktop machines, especially used for multimedia purposes and preferred in loading applications.



KINDS

- Blue Ray Disk – 40G
- Digital Versatile Disk
  - DVD-R – write once, 3.95G
  - DVD RW – rewritable, 3G
  - Single Layer and Double Layer
- Compact Disk
  - CD-R – write once, 650MB
  - CD-RW – rewritable, 700MB

Optical Drives

- CD-ROM — read CDs
- CD-Writer — read/write CDs
- DVD-Combo — read/write CDs, read DVD
- DVD Writer — read/write CDs
  read/write DVDs

Other Secondary Storage

- Solid-State Storage
  - No moving parts
  - Flash memory cards
  - USB flash drives

Address of the Memory

- Each cell in the memory has a unique physical address
- Most computers are byte-addressable
- n bits can address 2^n different locations

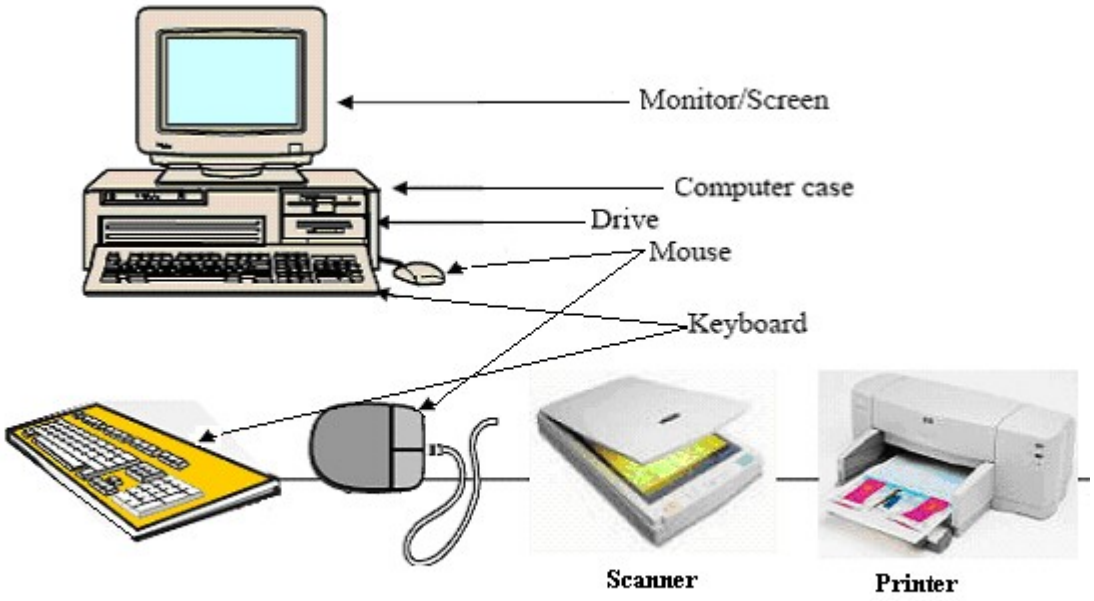
Address	Content
00000000	11100011
00000001	10101001
…	…
11111101	00000000
11111110	11111111
11111111	10100010

256 cell (byte) are addressed

Bus

- Transfers data or power between computer components or between computers.
- Logically connects several peripherals over the same set of wires.
- Each bus defines its set of connectors to physically plug devices, cards or cables together.

Input-Output Devices



Input Devices

- Allows data and programs to be sent to the CPU.
  - Keyboard
  - Mouse
  - Joystick
  - Microphone
  - Webcam
  - Scanner
  - Monitor

KEYBOARD

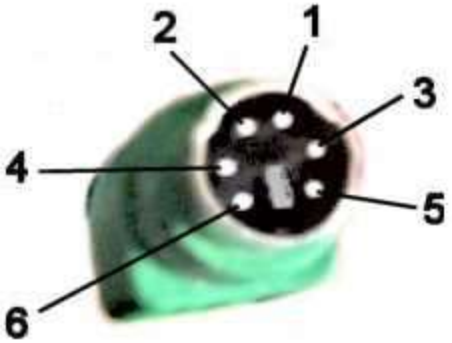
- Traditional keyboards
- Flexible keyboards
- Ergonomic keyboards
- Wireless keyboards
- PDA keyboards

Two Types of Mouse

- **Mechanical** - a type of computer mouse that has a rubber or metal ball on its underside and it can roll in every direction.
- **Optical:** This type uses a laser for detecting the mouse's movement.

How a Mouse Hooks Up to a PC

- PS/2 Mouse



- Serial Mouse

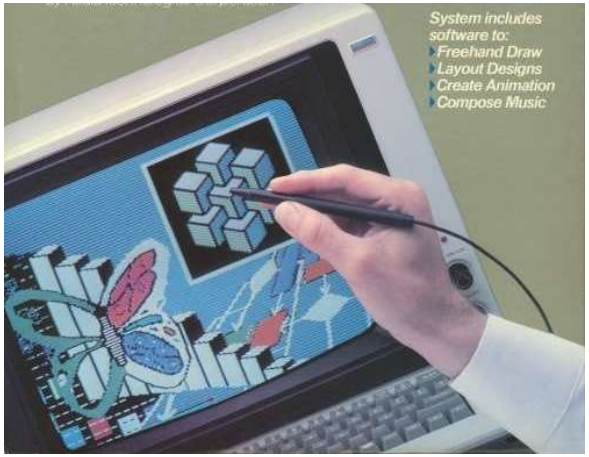
- USB/Cordless Mouse

Other Pointing Devices

- Trackball
- Track point
- Touch pad
- Touch Screen

- Joystick – input device for computer games

- Light Pens – light-sensitive penlike device



- Stylus – penlike device commonly used with tablet PCs and PDAs.

Scanning Devices

- Optical scanners
- Card readers
- Bar code readers
- Character and mark reco

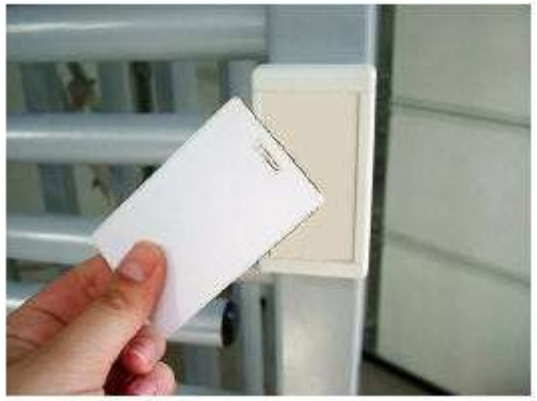


Image Capturing Devices

- Digital Cameras

- Digital Video Cameras

Output Devices

- Media used by the computer in displaying its responses to our requests and instructions.
- Monitor
- Audio Speakers
- Printer

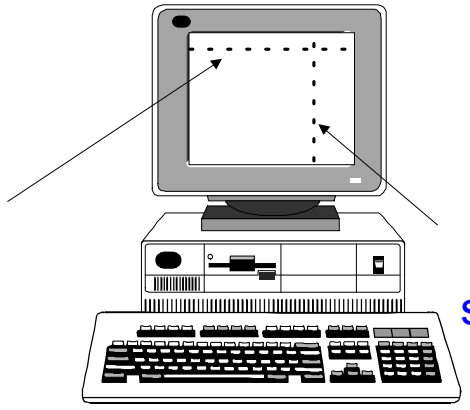
Types of Monitor

- Cathode Ray Tube (CRT)

- Liquid Crystal Display (LCD)

MONITOR

- Display device that operates like a television
  - Also known as CRT (cathode ray tube)
- Controlled by an output device called a *graphics card*
- Displayable area
  - Measured in dots per inch, dots are often referred to as pixels (short for picture element)
  - Standard resolution is 640 by 480
  - Many cards support resolution of 1280 by 1024 or better
  - Number of colors supported varies from 16 to billions



PRINTERS

- IMPACT PRINTERS uses pressure by physically striking the paper. Ex. Daisy wheel printers, line printers, dot matrix printers & band printers.
- NON-IMPACT PRINTER does not apply pressure on the paper but instead produces character by using lasers, ink spray, photography or heat.

Dot matrix printer

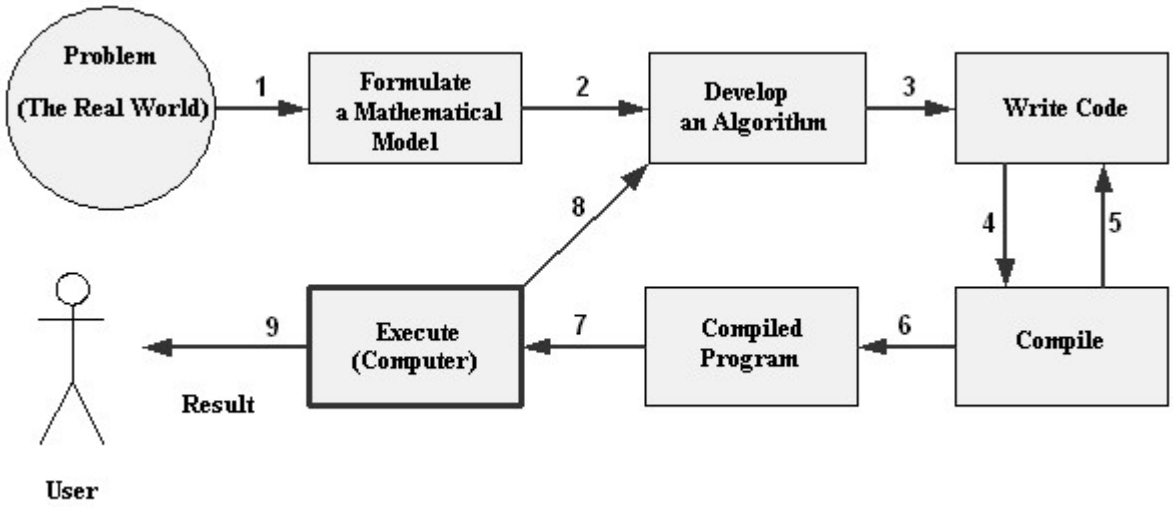
Inkjet printer

Laser printer

COMPUTER SOFTWARE

- Algorithms
- Programs and Programming Languages
- Classification of Computer Software: System Software and Application Software

THE PROCESS OF SOLVING PROBLEM WITH COMPUTERS



Problem Solving

- The art of finding a solution to a perplexing question
- How to solve it?
  - Understanding the problem
  - Devising a plan
  - Carrying out the plan
  - Looking back

COMPUTER PROBLEM-SOLVING

Three important phases:

- *Algorithm Development*: analyze, propose algorithm, test algorithm
- *Implementation*: code, test
- *Maintenance*:use, modify the program to meet chaining requirement or to correct errors.

What's an Algorithm?

- Shampoo Algorithm
  - Step 1: Wet hair
  - Step 2: Lather
  - Step 3: Rinse
  - Step 4: Repeat
- Is this enough information?

What's an Algorithm?

- Which step will be repeated? How many times do we need to repeat it?
- Shampoo Algorithm (Revision #1):

  Step 1: Wet hair

  Step 2: Lather

  Step 3: Rinse

  Step 4: Repeat Steps 1-4
- How many times do we repeat step 1? Infinitely? Or at most 2 times?

WHAT'S AN ALGORITHM?

- Keep a count of the number of times to repeat the steps
- Repeat the algorithm at most 2 times
- Shampoo Algorithm (Revision #2)

  Step 1: Set count = 0

  Step 2: Wet hair

  Step 3: Lather

  Step 4: Rinse

  Step 5: Increment count (increase its value by 1)

  Step 6: If count < 2 repeat steps 2-6

  Step 7: EXIT
- Will execute steps 2-6 **two** times

ALGORITHM

- Unambiguous instructions for solving a problem or subproblem in a finite amount of time using a finite amount of data

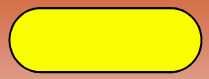
Algorithm Building Blocks

All problems can be solved by employing any one of the following building blocks or their combinations

1. Sequences
2. Conditionals
3. Loops

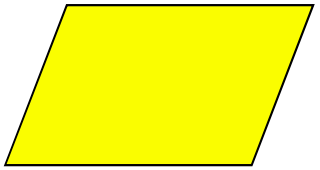
Review of Flowchart Elements

Start or stop

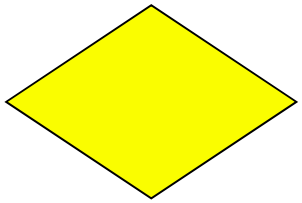


Process

Input or output



Decision



Flow line

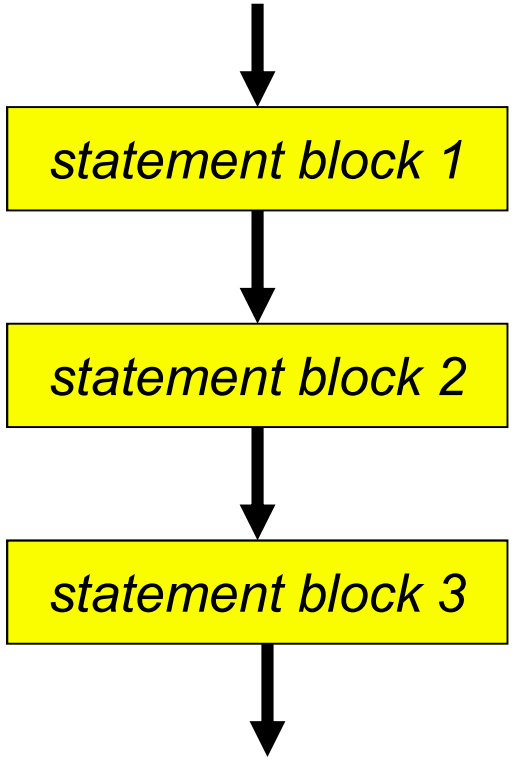
SEQUENCES

A sequence of instructions that are executed in the precise order they are written in:

statement block 1

statement block 2

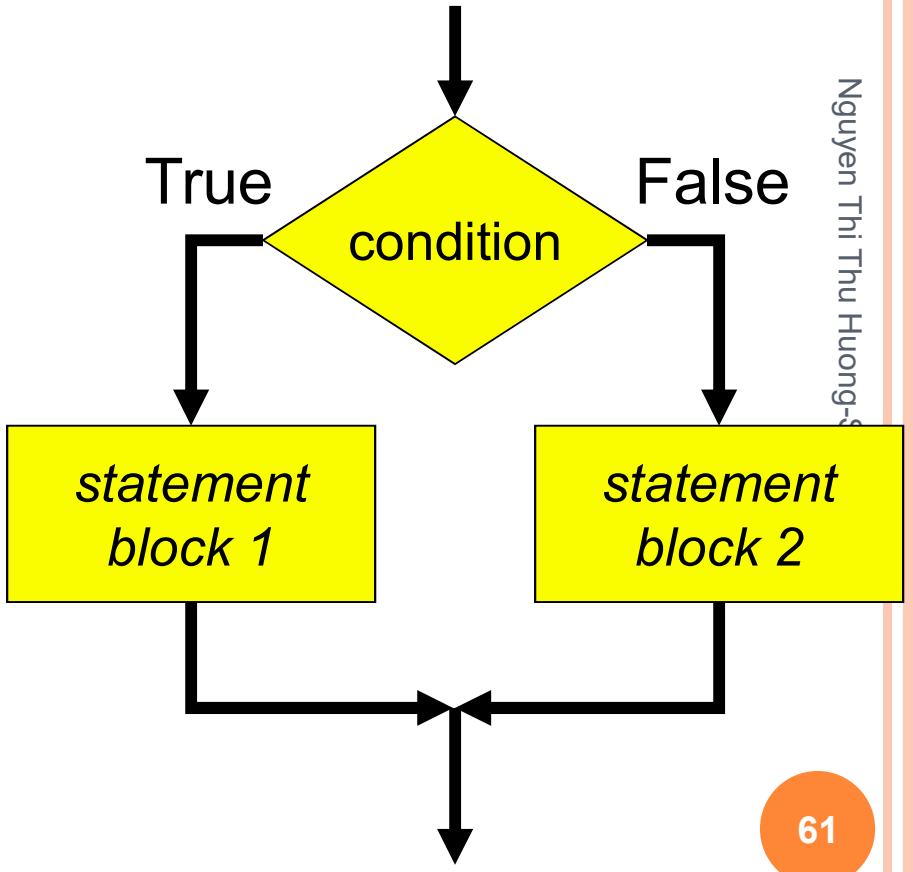
statement block 3



CONDITIONALS

Select between alternate courses of action depending upon the evaluation of a condition

```
If ( condition = true )
        statement block 1
Else
        statement block 2
End if
```



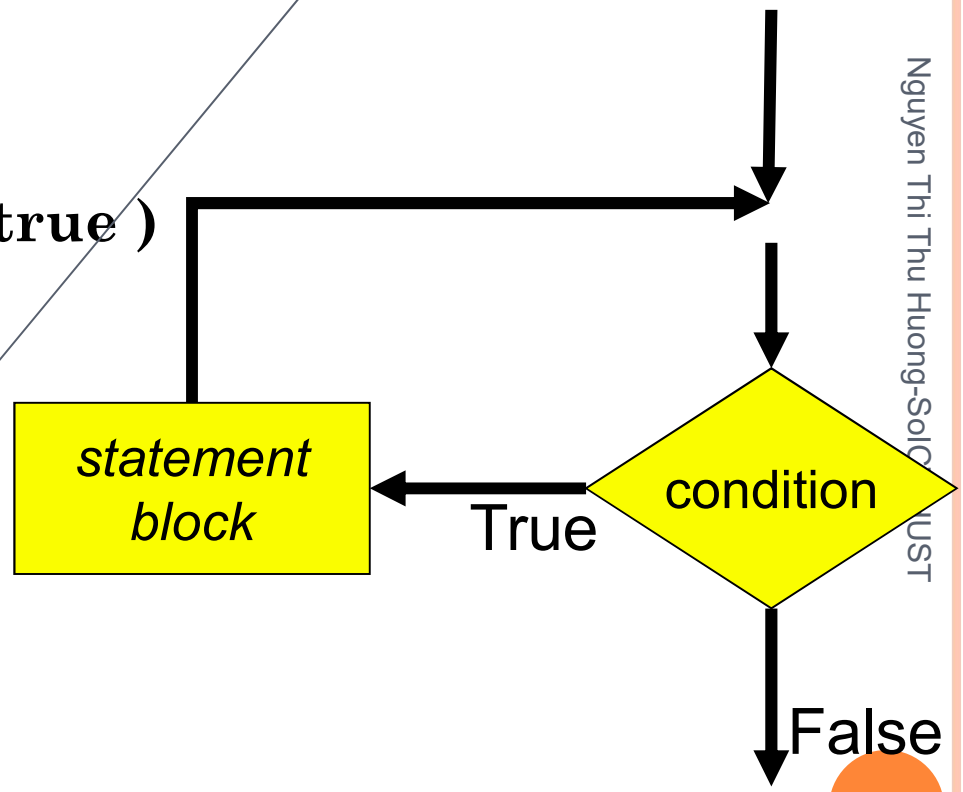
LOOPS

Loop through a set of statements as long as a condition is true

Loop while (condition = true)

statement block

End Loop



Problem Statement

Convert a decimal number into binary

Convert 75 to Binary

		remainder
2	75	
2	37	1
2	18	1
2	9	0
2	4	1
2	2	0
2	1	0
	0	1

1001011

Nguyen Thi Thu Huong-SoICT-HUST

Solution in Pseudo Code

1. Let the decimal number be an integer x, $x > 0$
2. Let the binary equivalent be an empty string y
3. Repeat while $x > 0$ {
   Determine the quotient & remainder of $x \div 2$
   y = CONCATENATE(remainder, y)
   x = quotient
   }
4. Print y
5. Stop

Q: Is this the only possible algorithm for converting a decimal number into a binary representation?

If not, then is this the best?

In terms of speed?

In terms of memory requirement?

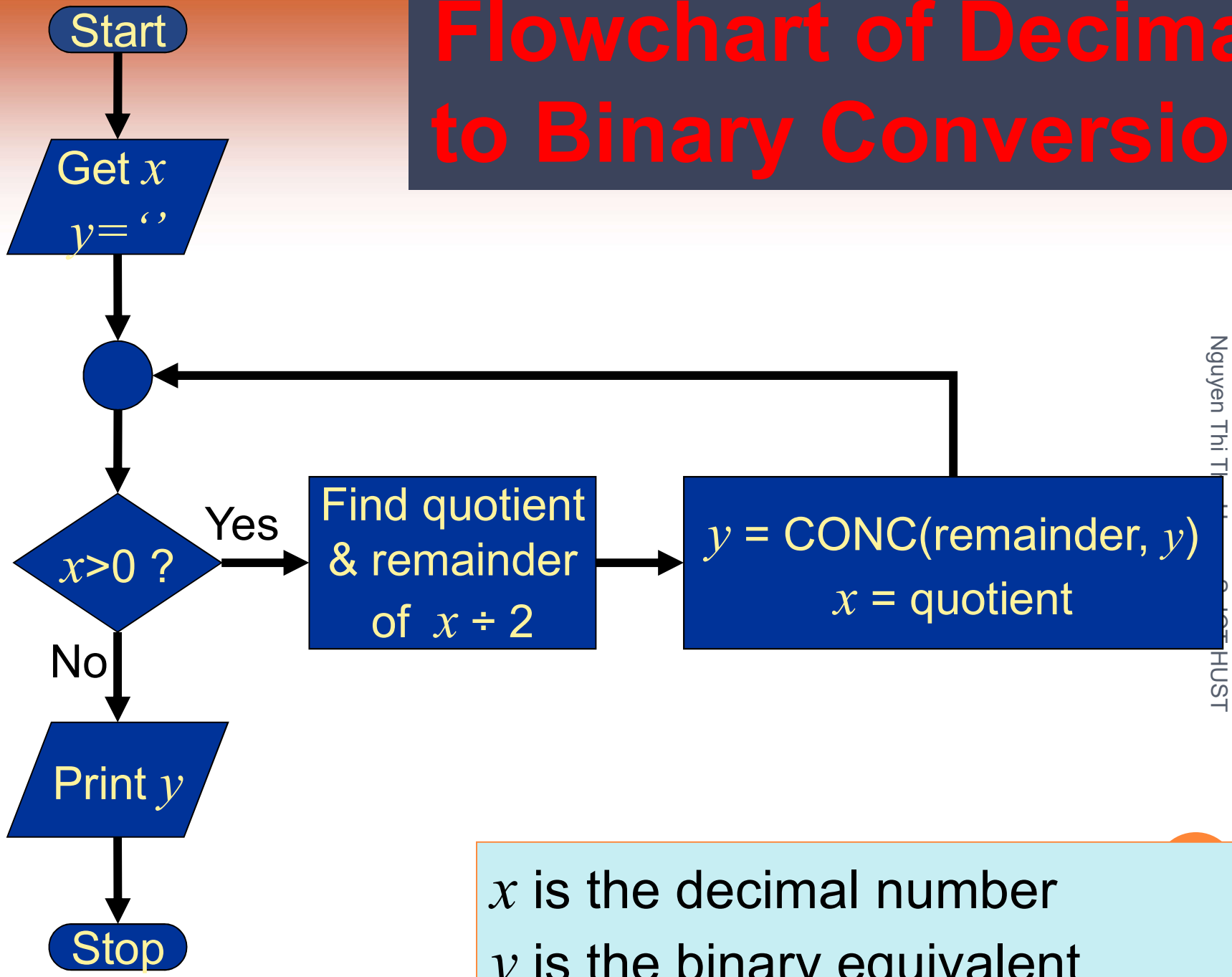
In terms of ease of implementation?

You must ask these questions after writing any algorithm!

Tips on Writing Good Pseudo Code

- Use indention for improved clarity
- Do not put "code" in pseudo code – make your pseudo code language independent
- Don't write pseudo code for yourself – write it in an unambiguous fashion so that anyone with a reasonable knowledge can understand and implement it
- Be consistent
- Prefer formulas over English language descriptions

Flowchart of Decimal to Binary Conversion



x is the decimal number

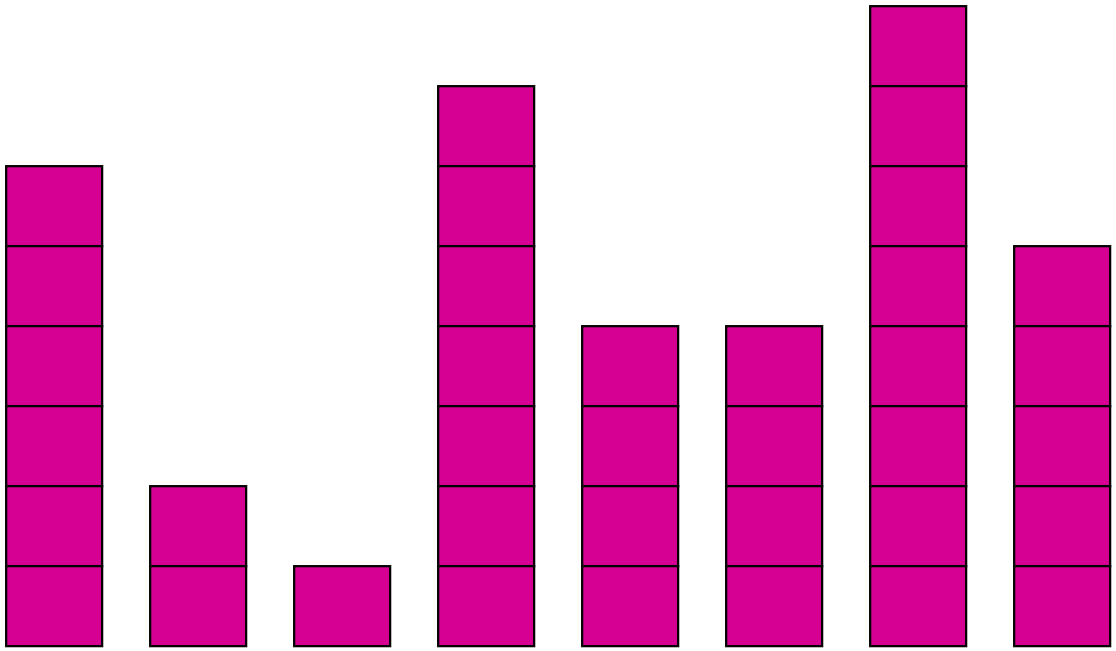
y is the binary equivalent

Questions

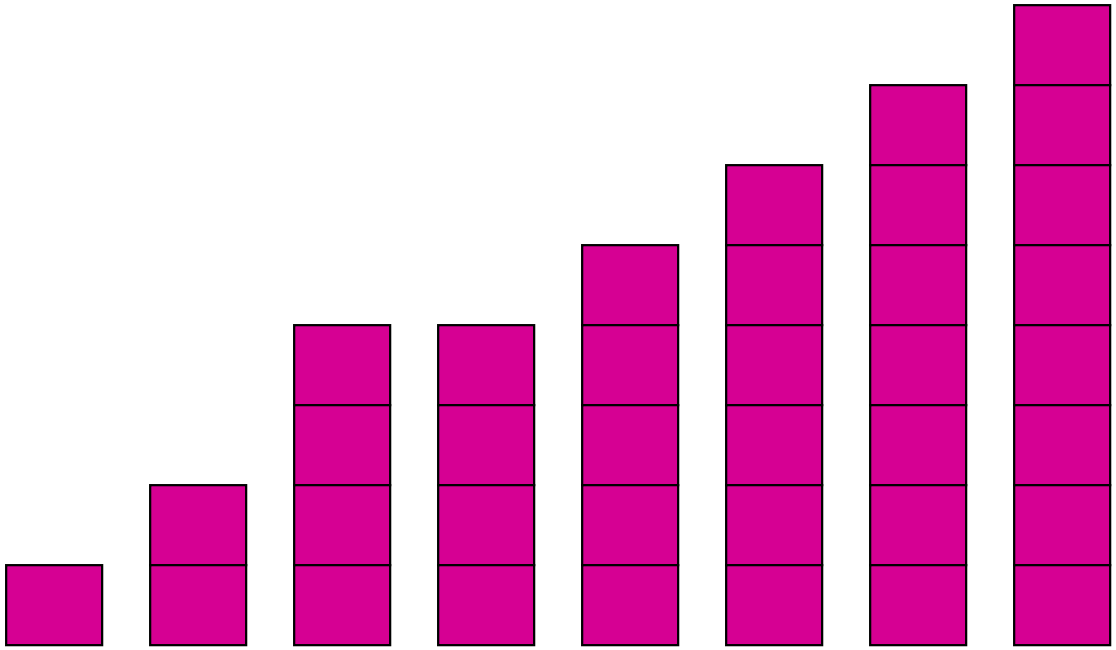
1. Does the flowchart depict the "correct" algorithm?
2. What do we mean by "correct", or better yet, what do we check for "correctness"?
3. One way is to check the algorithm for a variety of inputs
4. Does it perform satisfactorily for:
   - $x = 0$?
   - negative numbers?
   - numbers with fractional parts?

Another Example: Sorting

Sort the following objects w.r.t. their heights



Expected Result



STRATEGY

There are many strategies for solving this problem. We demonstrate a simple one:

Repeat the following steps while the list is un-sorted:

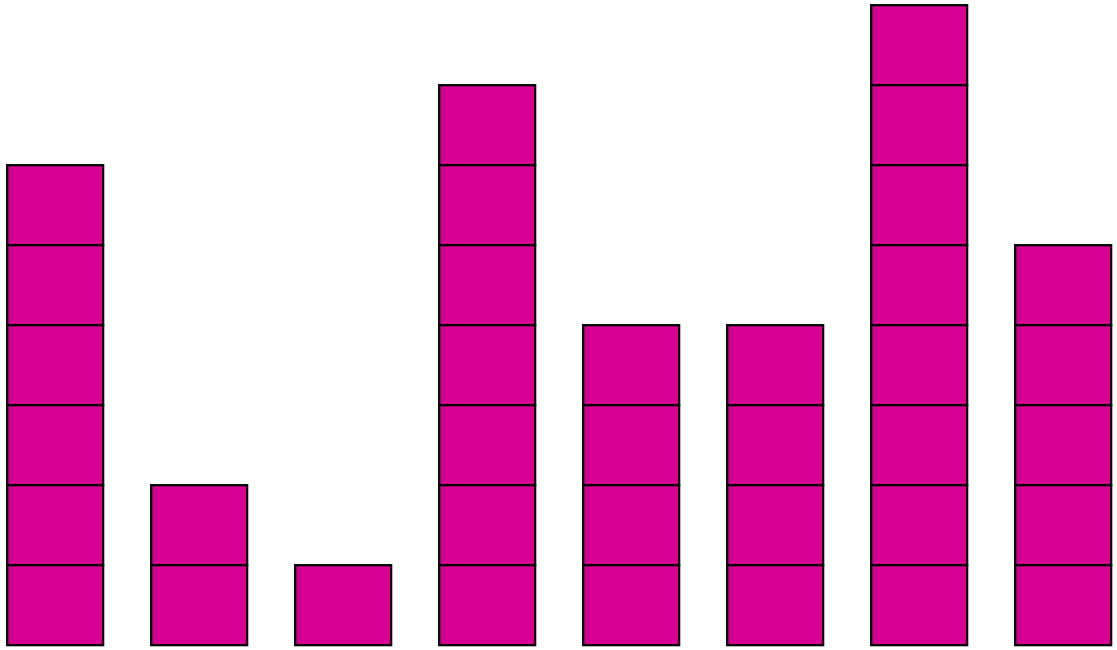
Start with the first object in the list

Swap it with the one next to it if they are in the wrong order

Repeat the same with the next to the first object

Keep on repeating until you reach the last object in the list

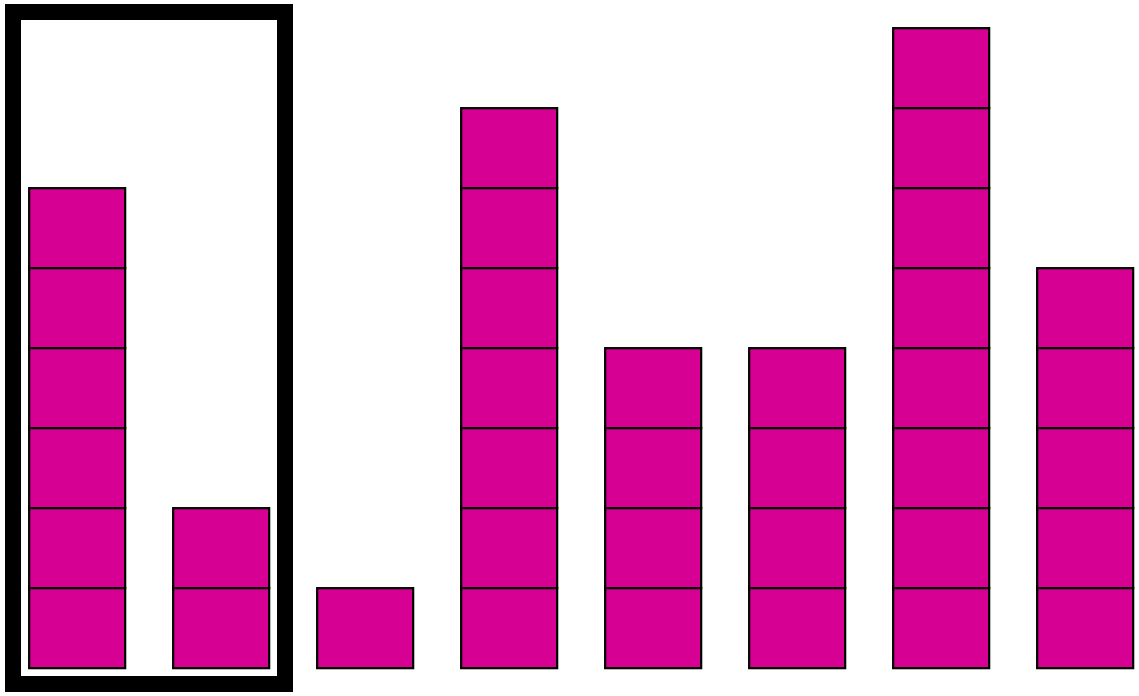
Back to the Objects to be Sorted



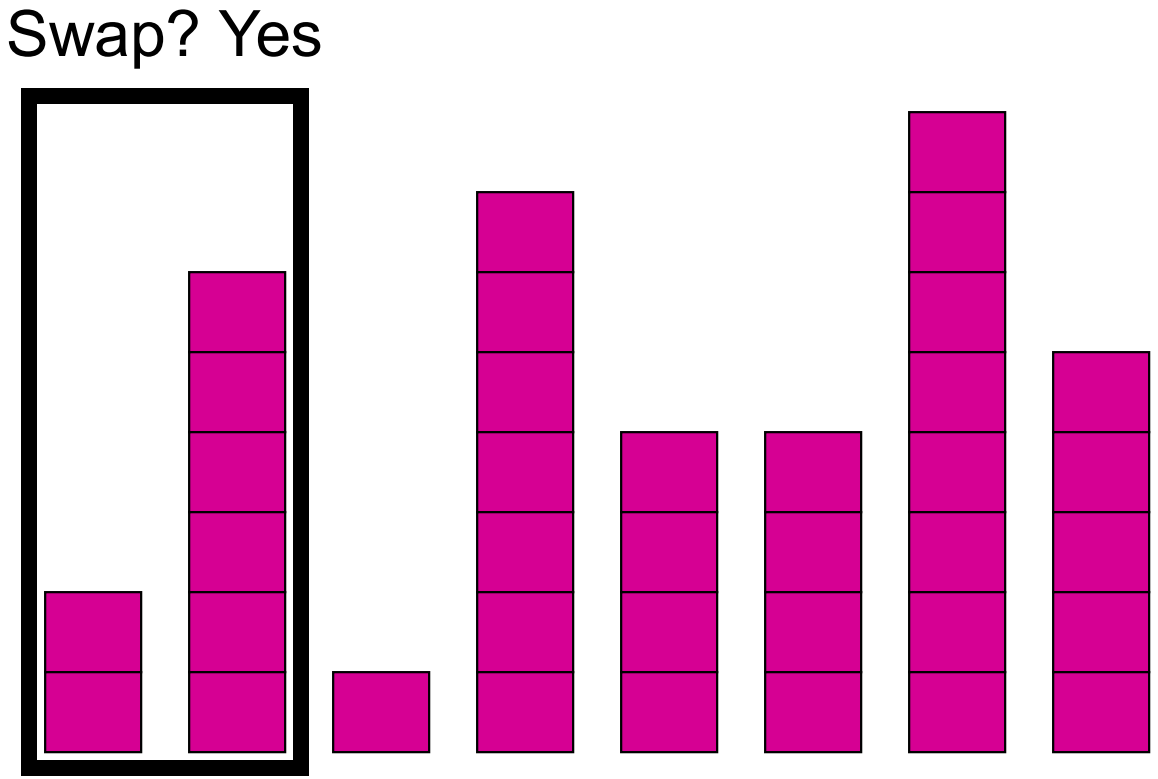
Q: Is the list sorted?

A: No

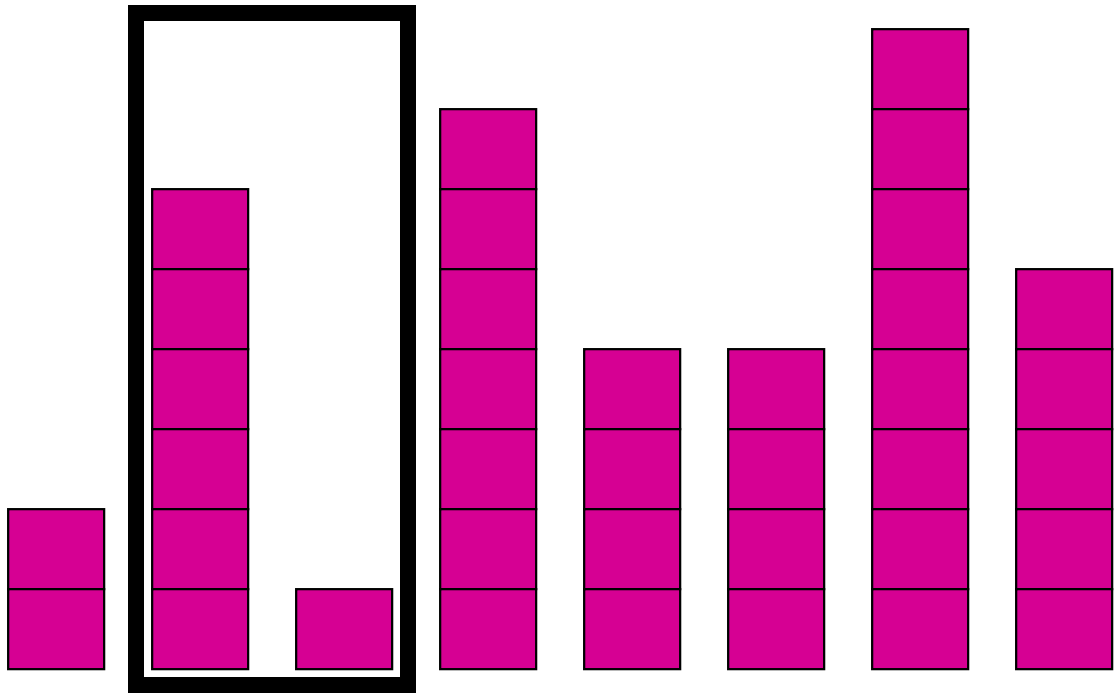
Sorting: Step A1



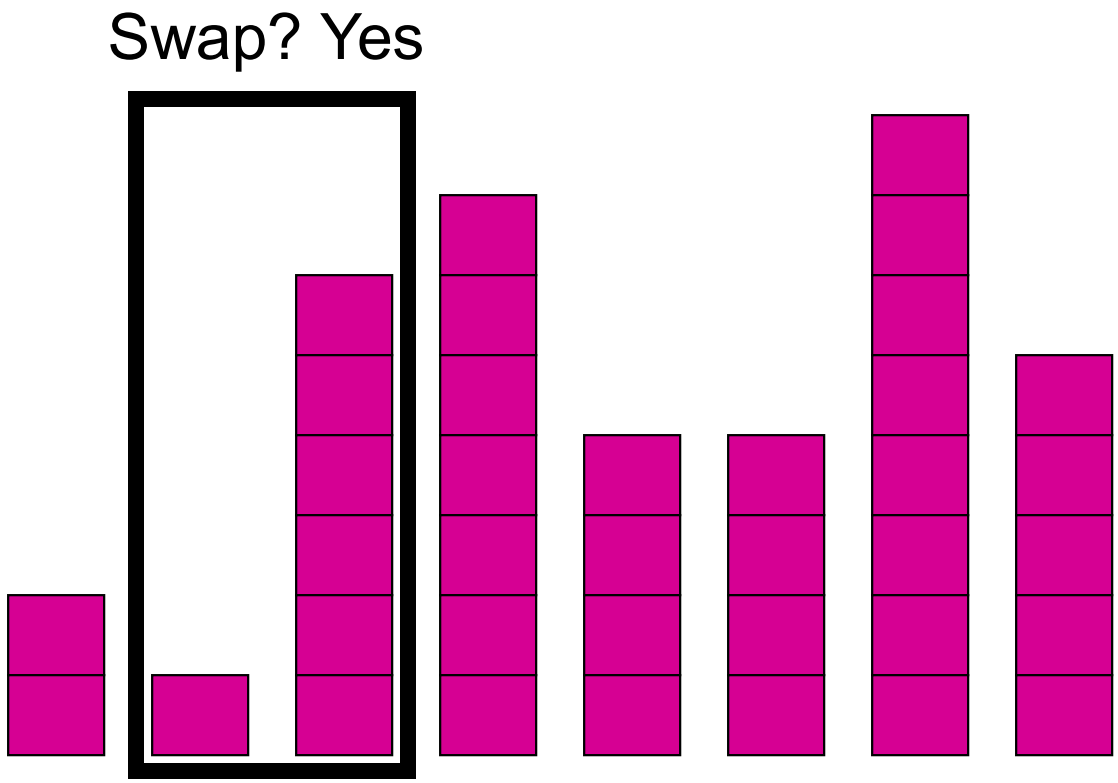
SORTING: STEP A1



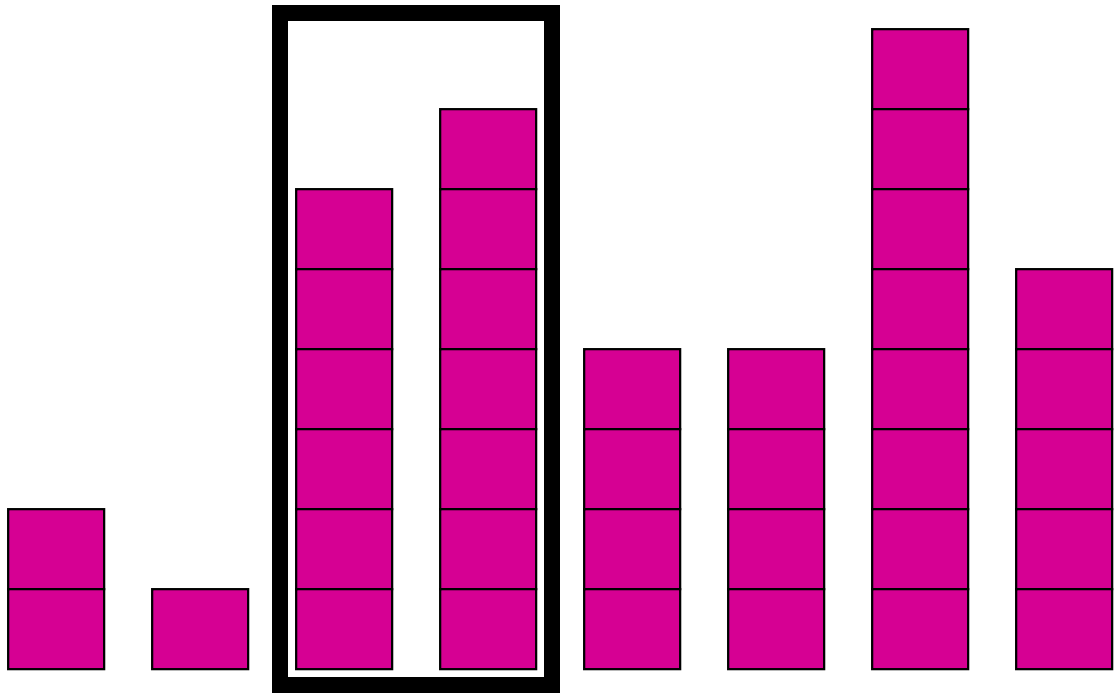
Sorting: Step A2



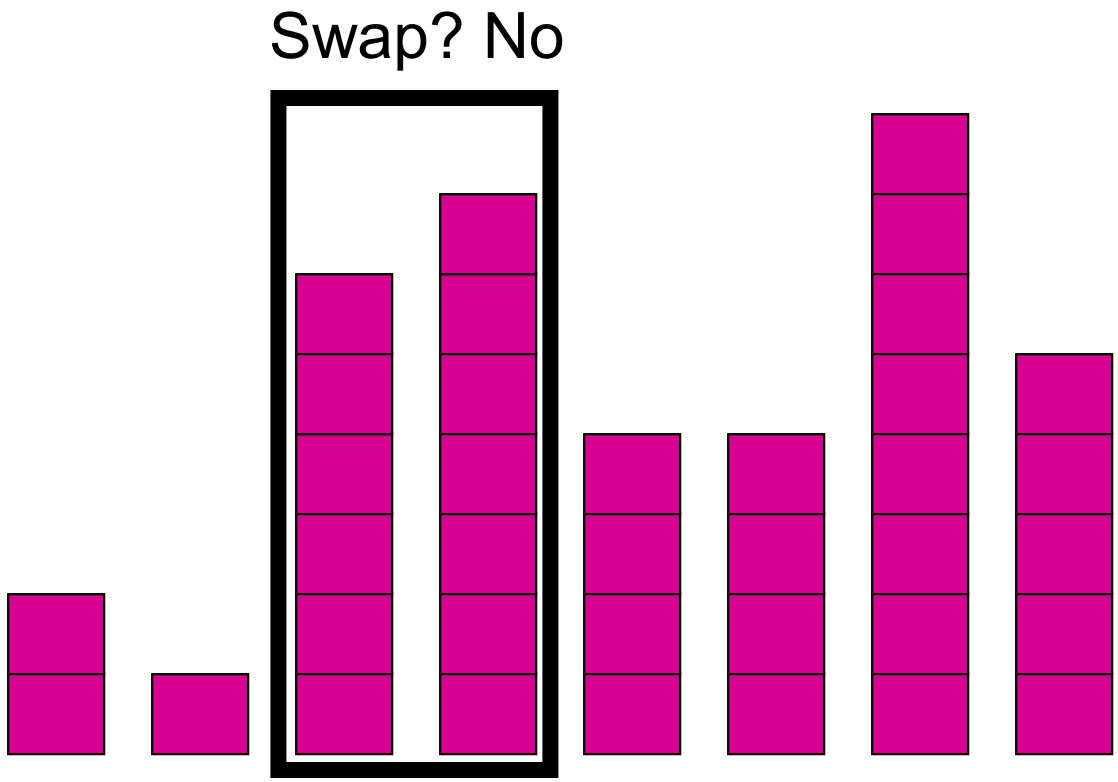
Sorting: Step A2



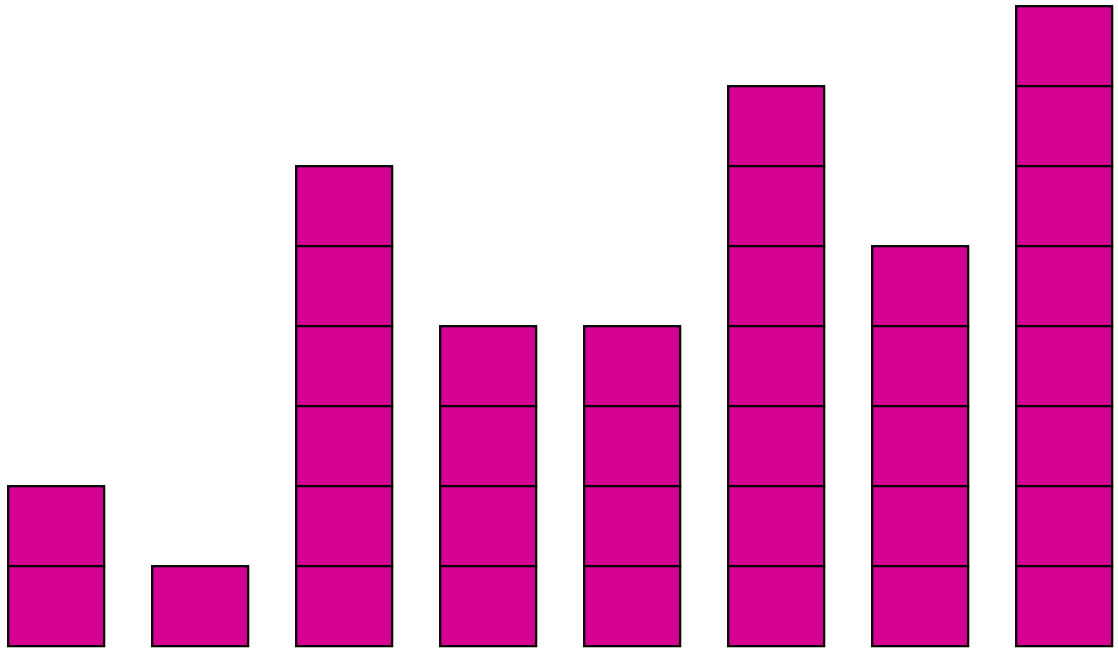
Sorting: Step A3



Sorting: Step A3



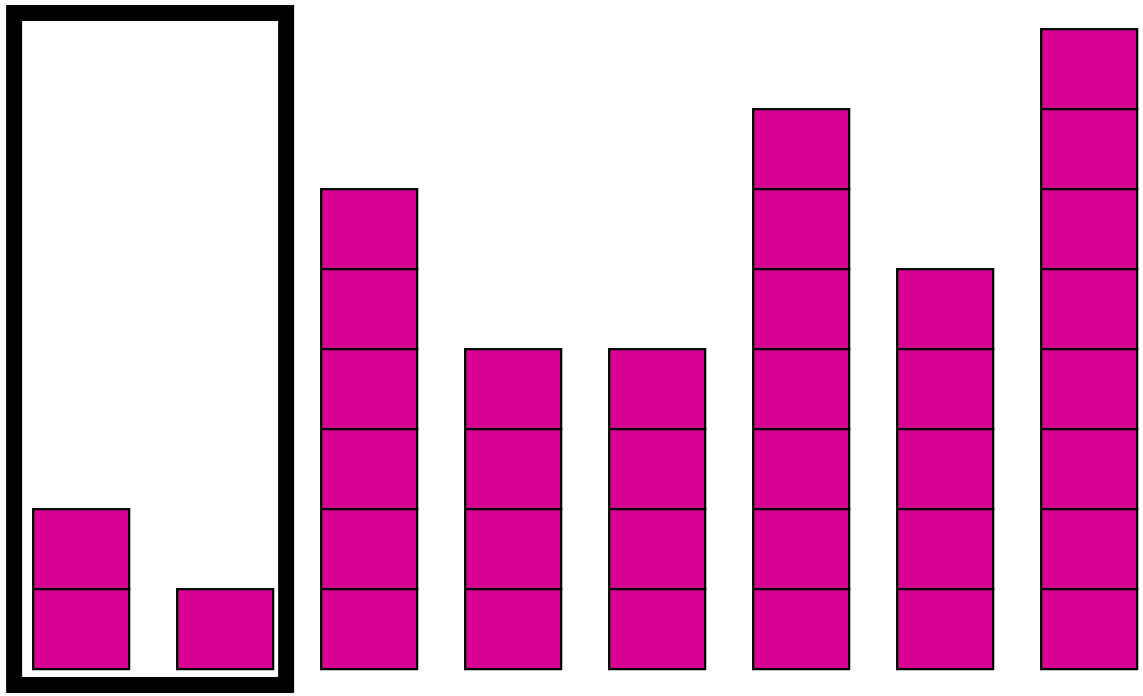
Sorting: After Step A7



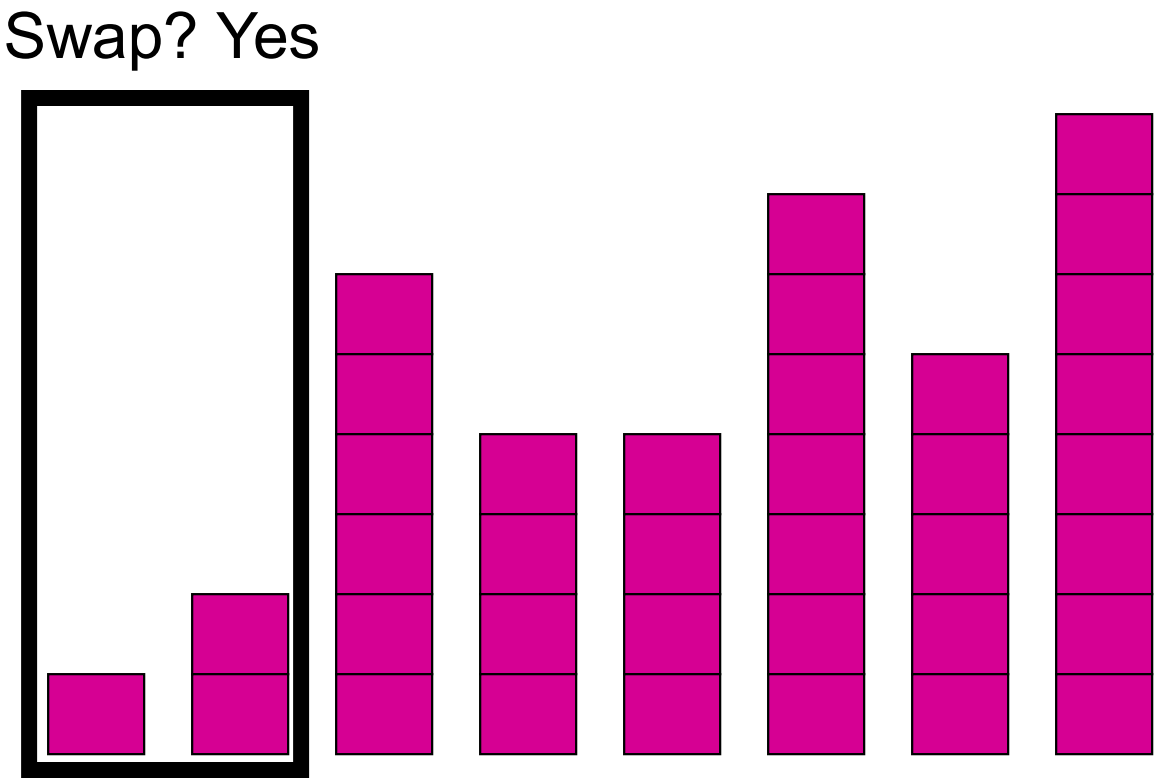
Q: Is the list sorted?

A: No

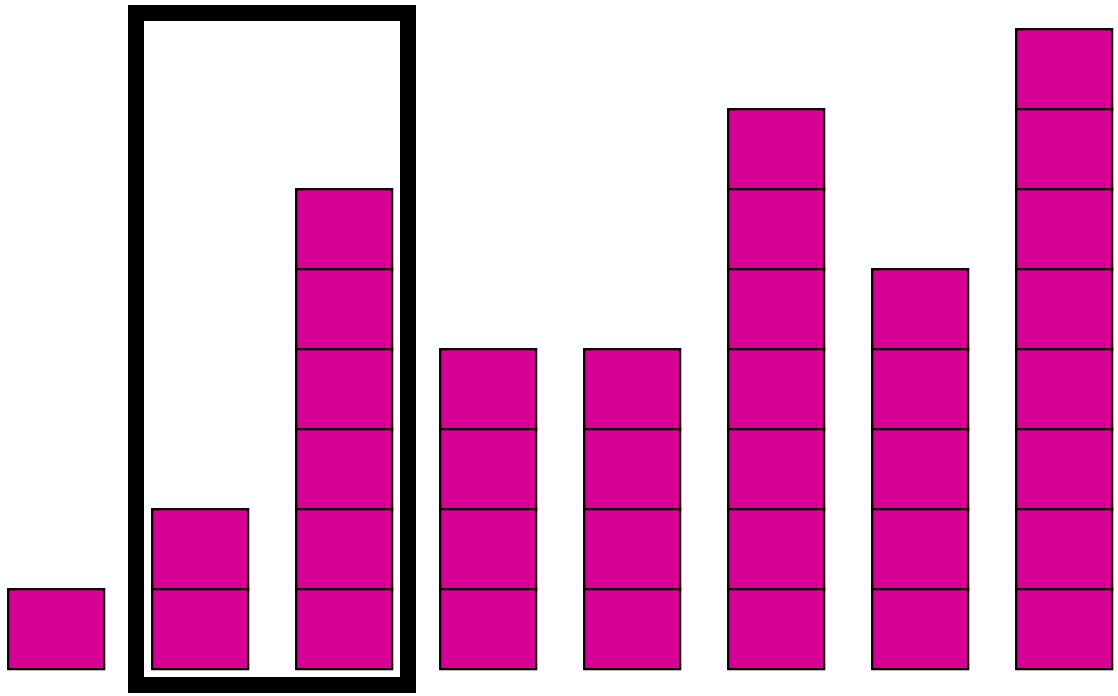
SORTING: STEP B1



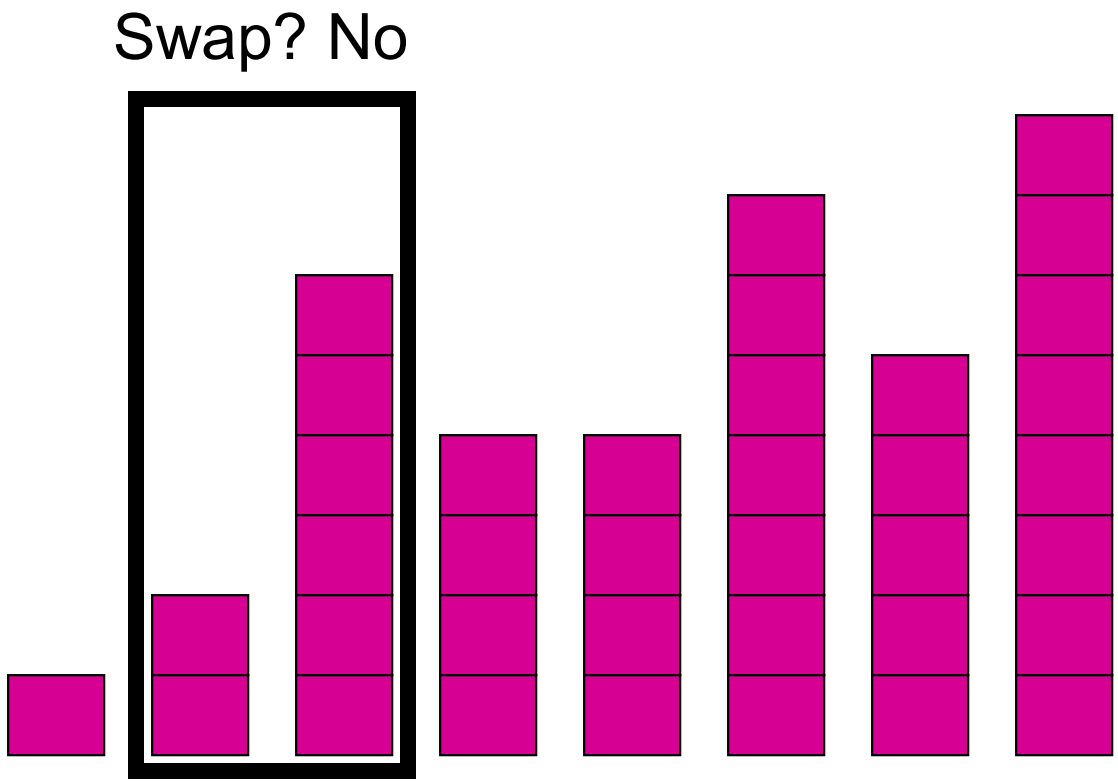
Sorting: Step B1



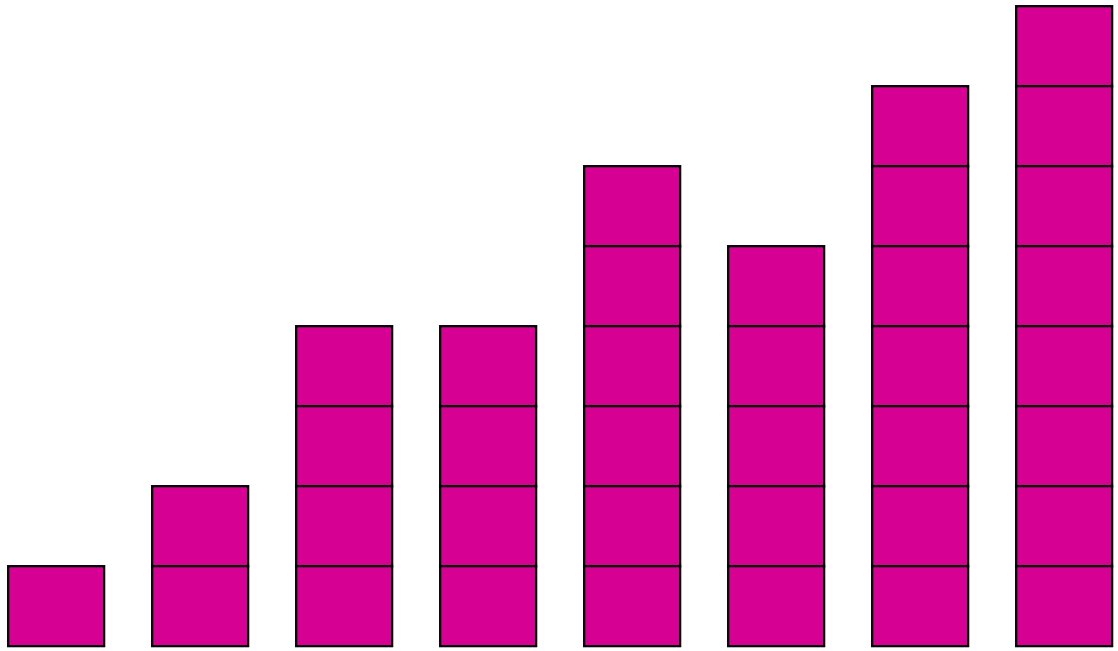
Sorting: Step B2



Sorting: Step B2



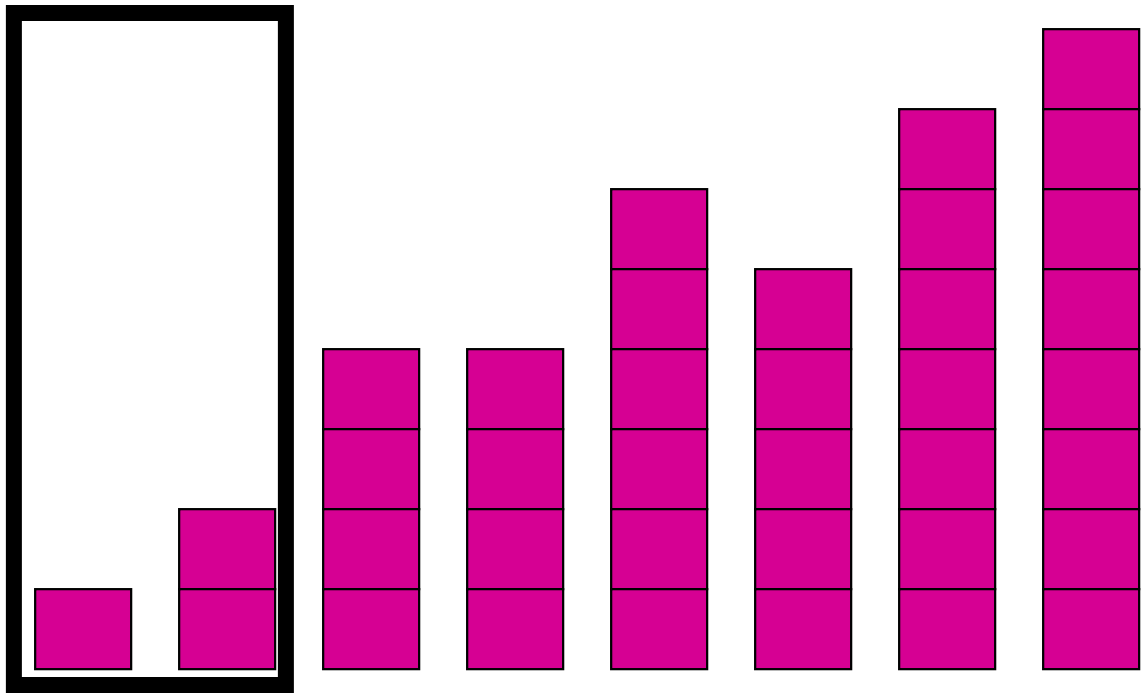
Sorting: After Step B6



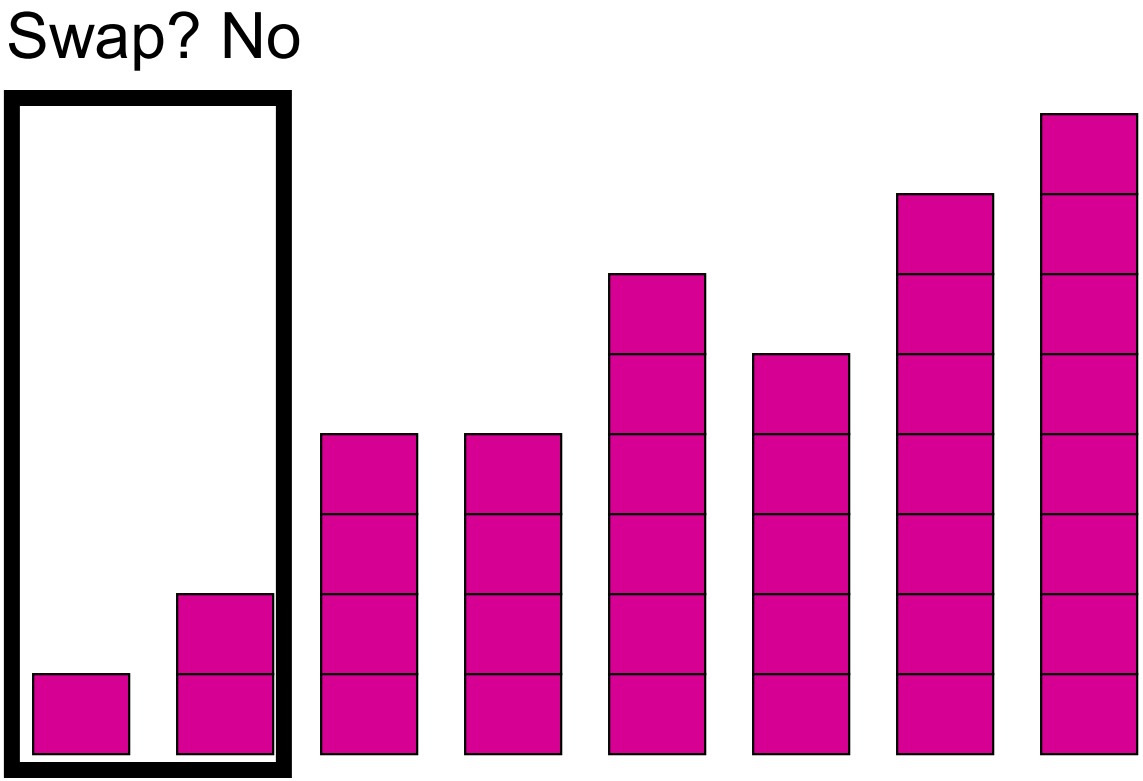
Q: Is the list sorted?

A: No

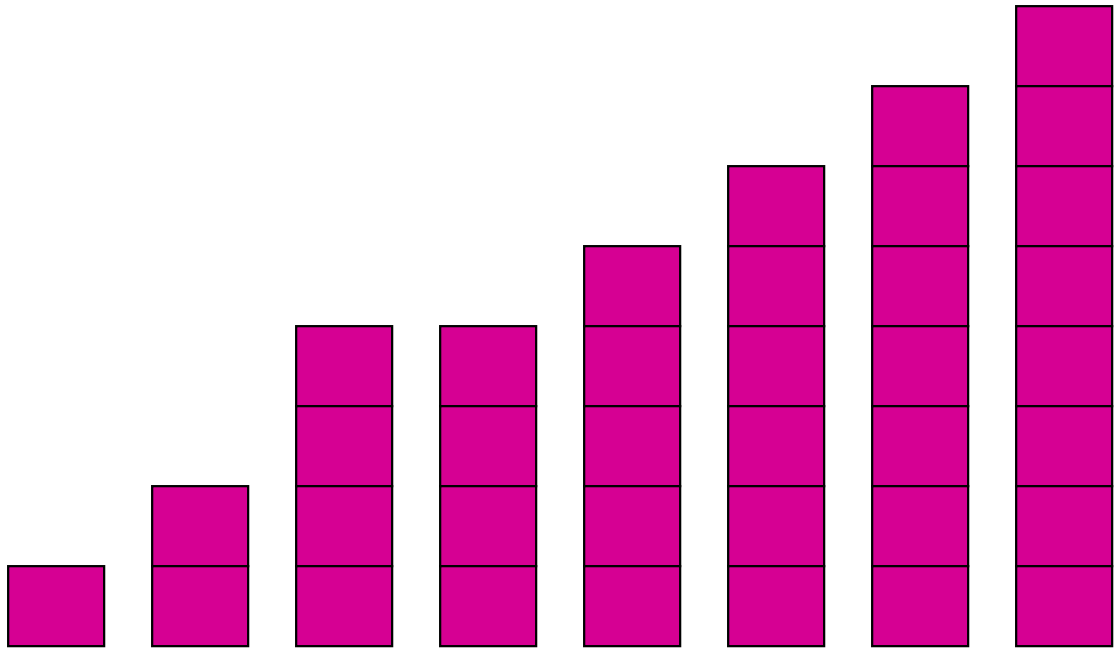
SORTING: STEP C1



SORTING: STEP C1



Sorting: After Step C5



Q: Is the list sorted?

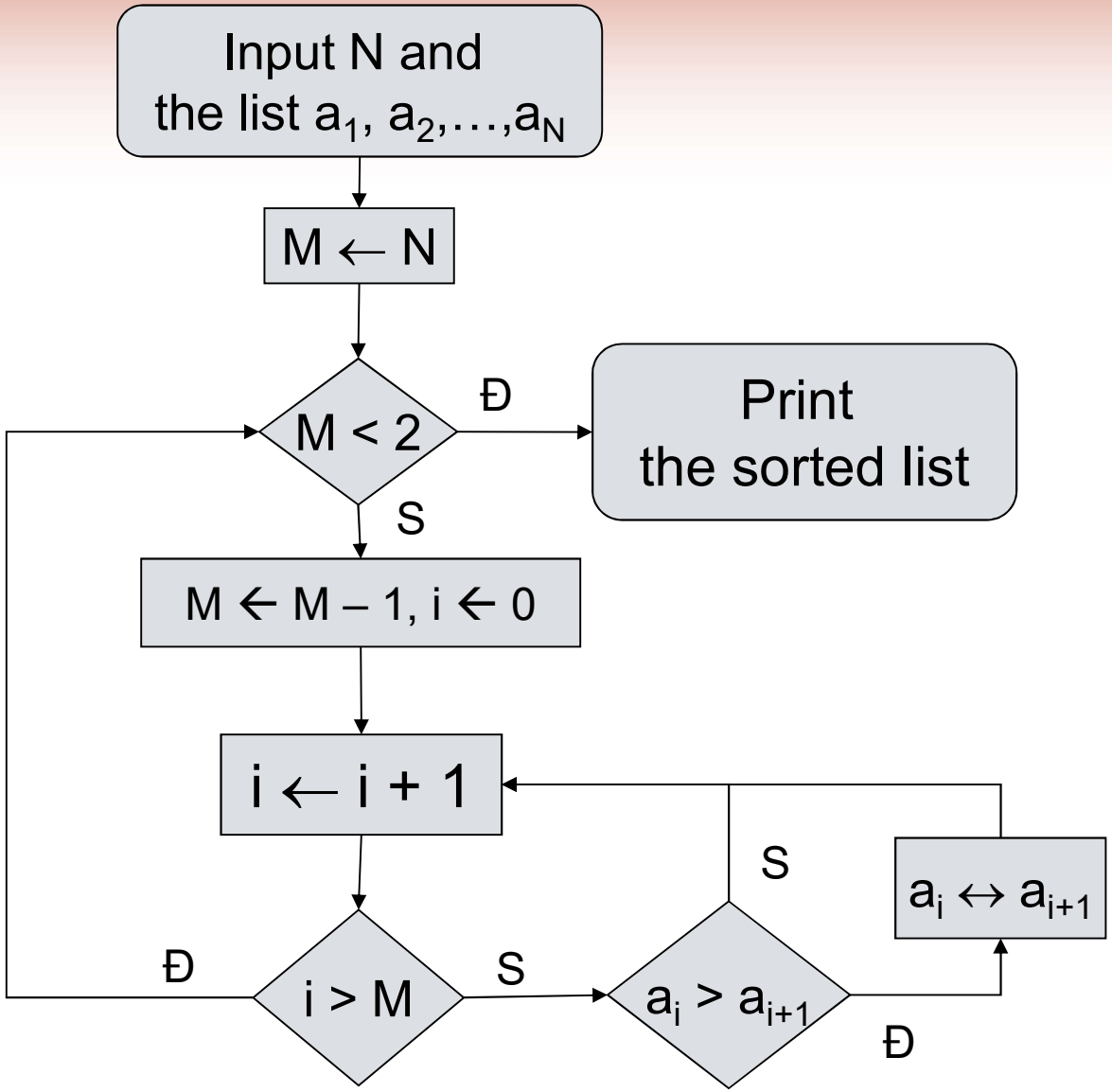
A: Yes

STOP

Flowchart for the Sorting Process

a is an array containing the heights

N is the total number of objects in the list



Is this the only possible algorithm for sorting a list?

Certainly not! In fact this one (called the "Bubble sort") is probably the worst (reasonable) algorithm for sorting a list – it is just too slow

You will learn a lot more about sorting in your future courses

Pros and Cons of Flowcharts

- Symbols are quite intuitive and almost universally understood
- Graphical nature makes the process of explaining an algorithm quite straightforward
- The process of writing an algorithm is too cumbersome
- Converting graphical form into code is not straight forward
- Another kind of flowcharts – Structured Flowcharts

Pros and Cons of Pseudo Code

- Closer in form to real code
- Use pseudo code as a starting point or outline for writing real code
- Converting each line into real code is not easy
- No standard rules exist for writing pseudo code

CHARACTERISTICS OF ALGORITHMS

- Exactness
- Effectiveness
- Guaranteed termination
- Generality

PROGRAM

- A sequence of instructions written to perform a specified task

PROGRAMMING LANGUAGE

- An artificial language that can be used to control the behavior of a machine, particularly a computer.
- Is defined through the use of syntactic and semantic rules
- Facilitate the task of organizing and manipulating information
- Express algorithms precisely.

MACHINE LANGUAGE

- System of instructions directly executed by a computer's CPU
- The lowest-level of abstraction
- Instructions are patterns of bits corresponding to different commands of the machine.

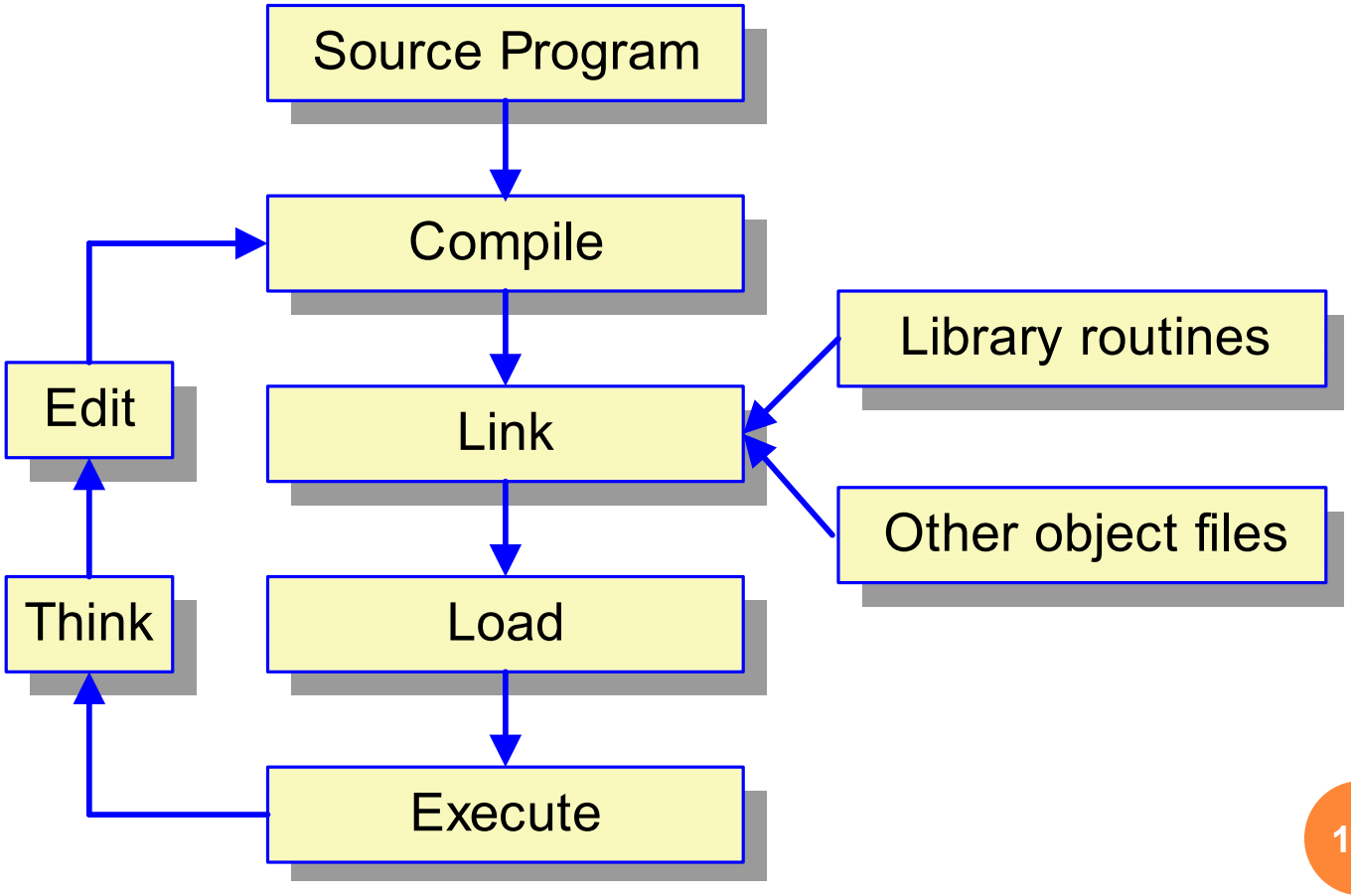
ASSEMBLY

- Low-level language for
- Implements a symbolic representation of the numeric machine codes of a particular CPU architecture.
- Is specific to a certain physical or virtual computer architecture
- Assembler : a program to translate assembly language statements into the target computer's machine code.

HIGH-LEVEL LANGUAGES

- More abstract, easier to use,
- More portable across platforms.
- Examples: Pascal, C, Visual Basic, SQL, ...
- Abstract away CPU operations
- They have been implemented by **translating** to machine languages

Software Development Cycle



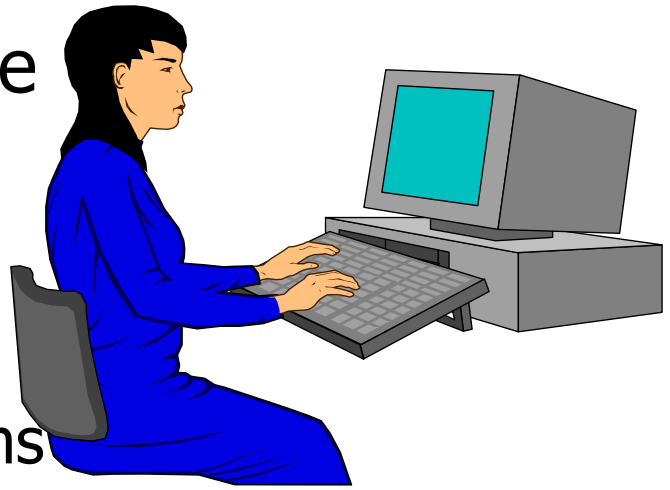
Classification of Computer Software

- Application software
  - Programs that help us to solve real-world problems
- System software
  - Programs that manage the computer system and interact with the hardware

Application Software

- Application software is the software that has made using computers indispensable and popular

- Common application software
  - Word processors
  - Electronic spreadsheet
  - Media players
  - Database management systems
  - Drawing programs



Disciplines of Computer Science

- Information Systems
- Artificial Intelligence
- Simulation

INFORMATION SYSTEMS

- Software that helps us manage and analyze data
- Data reflect almost every aspect of our lives
- Database : a structured set of data
- Typical tools to construct information systems : electronic spreadsheets , database management systems

Artificial Intelligence (AI)

- The study of computer systems that model and apply the intelligence of the human mind
- Aspects of AI
  - Knowledge representation
  - Expert Systems
  - Neural networks
  - Natural Language Processing
  - Robotics

SIMULATION

- Developing a model of a complex system and experimenting with the model to observe the result
- Examples:
  - Queuing Systems
  - Weather Forcasting
  - Computer Aid Design (CAD)
  - Embedded System

System Software

- Operating systems
- Translation systems
- Utilities

Translation System

- Set of programs used to develop software
- A key component of a translation system is a translator
- Some parts of translators
  - Compiler
    - Converts from one language to another
  - Linker
    - Combines resources
- Examples
  - Microsoft Visual C++, CBuilder
    - Performs compilation, linking, and other activities.

TYPES OF TRANSLATORS

- **Compiler** is a program that translate source code from a high-level programming language to a lower level language (e.g., assembly language or machine language)

- **Interpreter** is a program that translates and executes source language statements one line at a time

UTILITIES

- Virus Scanning software
- Backup Software
- Scan Disk
- Disk Defragmenter