ICAWMSCS ’24 May 27-31, 2024

High Performance Computing Workshop

By
Prof. Clement Onime
onime@ictp.it
ICTP,
Trieste, Italy

Contents

- Revision
  - Digital computers (history)
  - Basics of programming
  - Parallelism
    - What, why and how
  - Example of software code for basic algorithms

Overview

- Interactive classes (56%)
- hands-on sessions (44%)
  - Exercises
- Self motivation (100%)
  - Identify goals & targets
  - Write down your expectations..

Books & on-line reference materials

- Books
  - An Introduction to Parallel and Vector Scientific computing by Ronald W. Shonkwiler & Lew Lefton, Cambride texts in applied mathematics, 2006
  - Parallel Programming with MPI by Peter S. Pacheco, Morgan Kaufman Publishers, 1997
- On-line materials
  - Stanford University CS Education Library http://cslibrary.stanford.edu/
  - Lawrence Livermore National Laboratory https://computing.llnl.gov/?set=training&page=index
  - Durham University http://www.dur.ac.uk/resources/its/info/guides/

REVISION

Sectional Outline

- Computer hardware
- Programming languages
- Tools for programming
- Data types (integers & floats)
- Operators
- Statements
- File operations

Von Neumann Architecture

- General requirements for electronic computers (1945).
  - Main components: Memory, control unit, Arithmetic Logic Unit (ALU) and input/output unit
  - Classically used for designing all computers/CPUs
- Proposed stored/changeable programs executed sequentially.

Who was John Von Neumann?

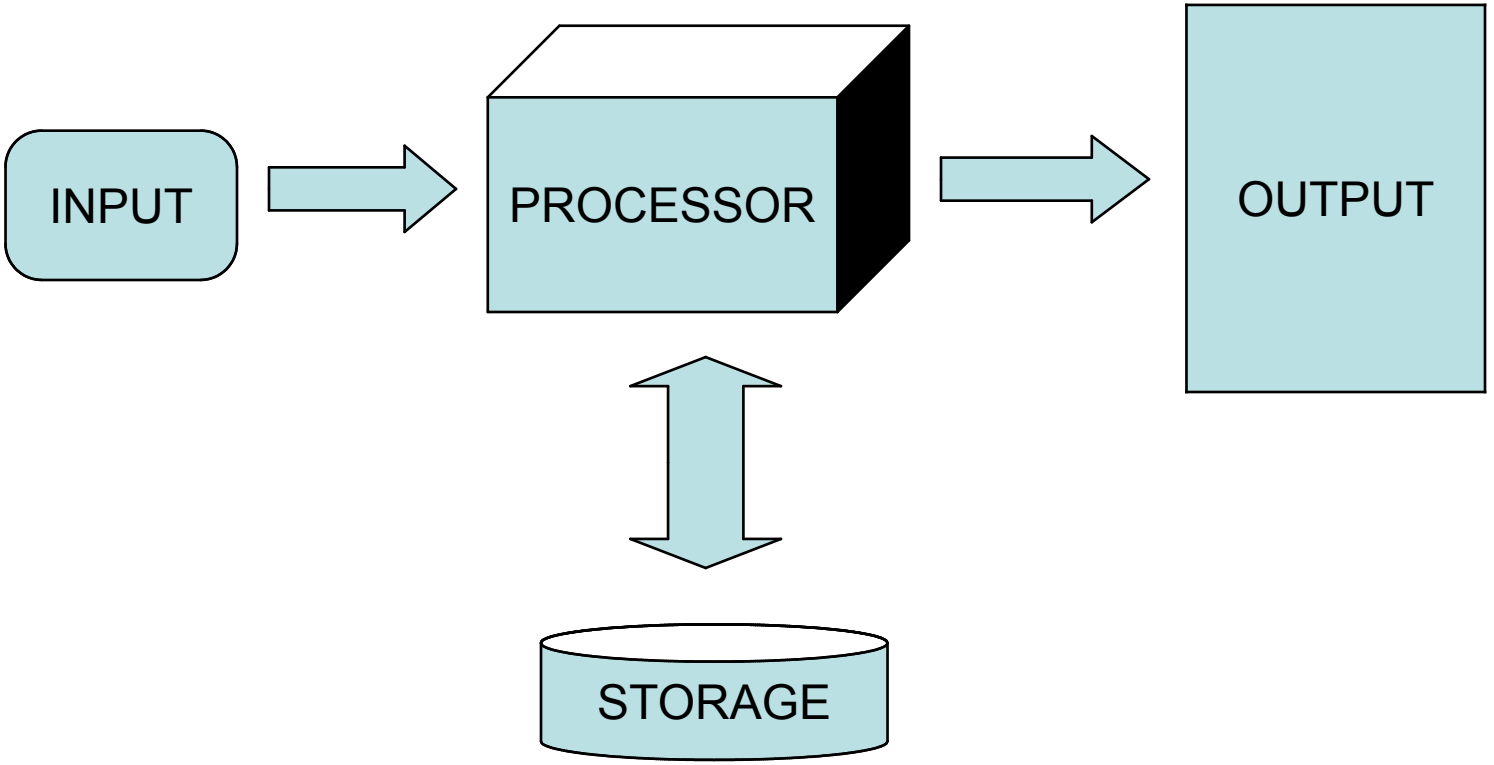
Class Exercise:

- Who was Von Neumann?
  - Mathematician
- What are some of his notable inventions or work?
  - Von Neumann architecture

What is a digital computer?

- Our starting definition is that a digital computer is a device that performs computations or calculations based on digits (numbers).
- Digital computers have four main parts:
  - Central Processing Unit (Processor)
  - Storage (memory)
  - Input
  - Output
- Before 1946 all digital computers worked internally with base 10 digits. The modern digital computers however work internally with in base 2 digits.

Block diagram of a digital computer



History of Digital computers

1645 - 1964

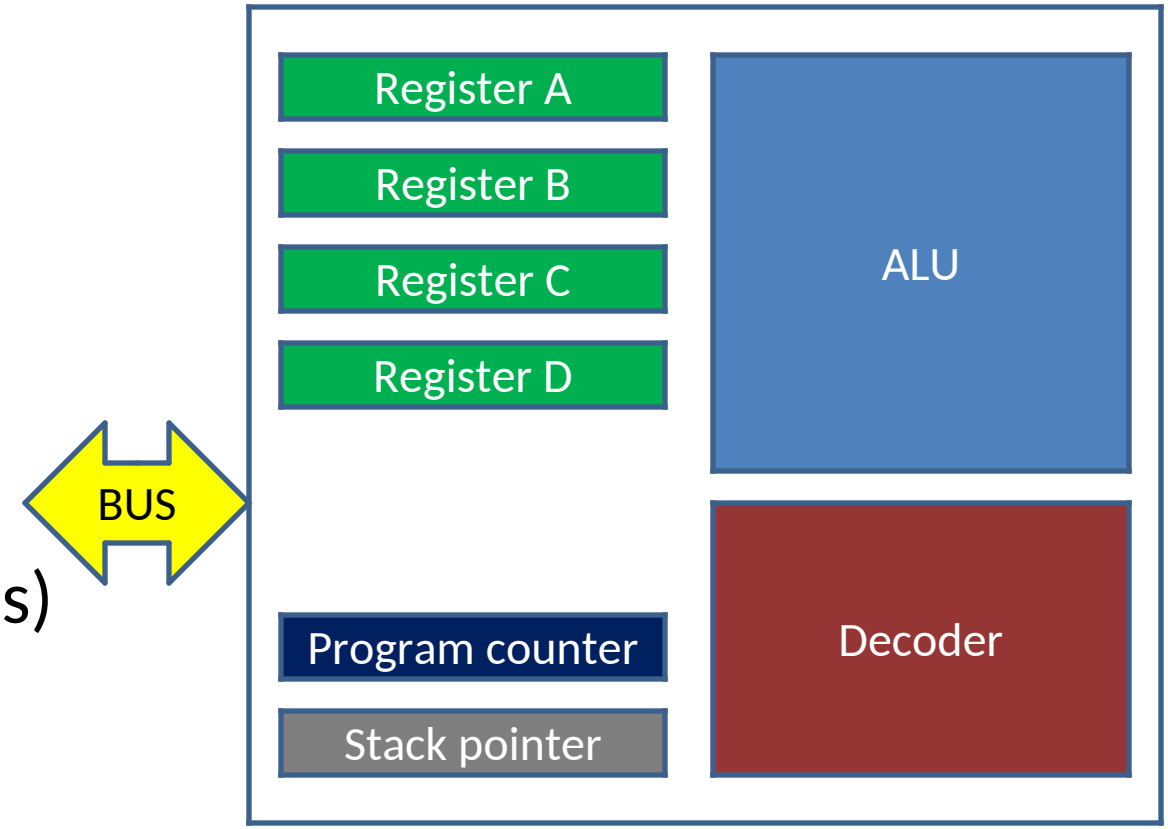
- Mechanical digital computers
  - Pascal's calculator (1645 - 1694)
  - Stepped Reckoner (1694 - 1851)
  - Arithmometers
  - comptometers
  - Comptographs
- Electromechanical digital computers
  - Sumlock comptometer Ltd, UK
    - Anita mark VII, VIII, 10 and 11
- Electronic
  - Sharp Corporation Japan (1964)
    - all transistor device..

Key facts Digital computers

- Factors
  - Solve challenge of performing computations
  - Reduce human errors
  - Improve speed
- Notes
  - Pascal (1645)
    - Patent first
  - Libnitz (1694)
    - Wheel technology was used for about 200 years (open source)
  - Thomas 1820
    - Patent first
    - Build design, quality, reliability, accuracy
    - provided user manual

Microprocessor Internals

- Main parts
  - CPU
  - Memory
  - Data bus
  - Clock
- Bottleneck(s)



Overview of Microprocessor

Exercise

- What is your motivation?
  - Chemistry:
  - Physics:
  - Engineering:

Storage sizes in Computing

Item	Size in computing	Commercial size
Kilobyte	$2^{10} = 1024$	$10^3 = 1000$
Megabyte	$2^{20} = 1024^2 = 1{,}048{,}576$	$10^6 = 1{,}000{,}000$
Gigabyte	$2^{30} = 1024^3 = 1{,}073{,}741{,}824$	$10^9 = 1{,}000{,}000{,}000$
Terabyte	$2^{40} = 1024^4 = 1.09951162778e+12$	$10^{12} = 1{,}000{,}000{,}000{,}000$
Petabyte	$2^{50} = 1024^5 = 1.12589990684 \times 10^{15}$	10^{15}
Exabyte	**$2^{60} = 1024^6 = 1.15292150461 \times 10^{18}$**	**10^{18}**
Yottabyte	$2^{70} = 1024^7 = 1.18059162072 \times 10^{21}$	10^{21}
Zettabyte	$2^{80} = 1024^8 = 1.20892581961 \times 10^{24}$	10^{24}

Programming languages

- Native/executable computer/machine language
  - Assembly code
- Human readable languages
  - Interpreted (single executable with different program file as input)
    - Perl, python or tcl
    - matlab, octave, etc
  - Compiled (transforms readable input into executable code
    - Fortran & C
  - Java is semi compiled into byte-code

Useful tools in programming

- Creating and modifying program source code
  - Editors (may include syntax highlighting checking)
- Translating source code into machine or byte code
  - Compilers (identify & highlight errors in source code)
- Tracing execution or runtime code for errors
  - Debuggers
- Integrated Development Environments
  - Combine all above tools in one interface.
- Revision Control Systems
  - subversion, git, etc

Computer Data types - Integers

- Integer
  - Whole numbers, typically signed representations
  - Char (8 bit)
  - Short (16 bit)
  - Int : 16 or 32bit
  - Long (32 bit)
  - Long long (64 bits)
  - Notes:
    - Unsigned types basically double the maximum possible positive value e.g unsigned short is $0 < x < 65536$

Computer Data types - Floats

- Numbers having two parts separated by a decimal point
- Single precision (32 bit, 6 digits after decimal)
- Double precision (64 bit, 15 digits after decimal)
- Long double (80 bit, even bigger)
  - Standard is double, float(single) may be used to save memory
  - Always avoid equality tests

$$(-1)^{sign} * 2^{exponent-bias} * 1.mantissa$$

$$(-1)^{sign} * 2^{1-bias} * 0.mantissa$$ *(this avoids underflow)*

$$(-1)^{sign}\left(1+\sum_{i=1}^{23} f_{23-i}2^{-i}\right)2^{e-127}$$

$$(-1)^{sign}\left(1+\sum_{i=1}^{52} f_{52-i}2^{-i}\right)2^{e-1023}$$

Operators

- Arithmetic: (/ * + - %)
- Relational: (== != > < <= >=)
- Logical: (!! && !)
- Binary: (^ ~ & |)

*What about += *= /= and -= ?*

Statements

- Executable
- Non-executable
- Control
- Input/output
- Built-in functions
- Functions
- Subroutines
- Logical

- *Exercise - Classify the following*

  X = 42
  break
  If ( )
  fopen
  printf
  Sqrt
  x < 5
  *Char * memcpy()*
  &&
  !

File operations

- Input
  - Sometimes can differentiate between text and binary files
  - Sequential or random
- Output
  - Create non existing
  - overwrite or clobber existing
  - Or append
- Fopen, printf, scanf, fgetc, fclose, fputc, ungetc