Introduction to Computer Systems (1)

Basic definitions and history of the Computer Systems

Piotr Mielecki Ph. D.

http://www.wssk.wroc.pl/~mielecki

piotr.mielecki@pwr.edu.pl

pmielecki@gmail.com

Literature

1. Stallings W., Computer Organization and Architecture: Designing for Performance.
2. Website of this handbook: http://www.prenhall.com/stallings/details1.html
3. Wikipedia

Definitions

What the Computer is?

- A computer is a machine which manipulates data according to a list of instructions which makes it an ideal example of a data processing system. This definition is true for most of the today's computers, based on von Neumann's concept. But it doesn't cover some more progressive concepts (like neural networks which don't have any lists of instructions, for example).
- We can also define the computer as "the machine which automatically processes the information". This definition seems to be less precise, but it's closer to the main reason for building the computers.
- The English word "compute" means "calculate", "count" etc. In fact computers can also manage data in other ways than only calculating the mathematical formulas (word processing, collecting data in databases, making communication, driving other machines like airplanes, laboratory equipment, lathes etc.).

Definitions

What the Computer System is?

- A computer system is a set of 3 elements:
  - HARDWARE – this term covers all of those parts of a computer that are tangible, material objects. Circuits, displays, power supplies, cables, keyboards, printers and mice are all hardware.
  - SOFTWARE – this is a general term used to describe a collection of computer programs, procedures and documentation that perform some task on a computer system. The term includes application software such as word processors which perform productive tasks for users, system software such as operating systems, which interface with hardware to provide the necessary services for application software, and middleware which controls and co-ordinates distributed systems.
  - USER – usually a person, sometimes the abstract object (like the anonymous account, on behalf of which such a program like HTTP server runs on Internet server) which uses the hardware and software to complete actual tasks.

Definitions

How we are describing the computer systems?

- Architecture is those attributes visible to the programmer (like instruction set, number of bits used for data representation, I/O mechanisms, addressing techniques etc.).
- Organization is how features are implemented (control signals, interfaces, memory technology etc.).
- Structure is the way in which components relate to each other.
- Function is the operation of individual components as part of the structure.

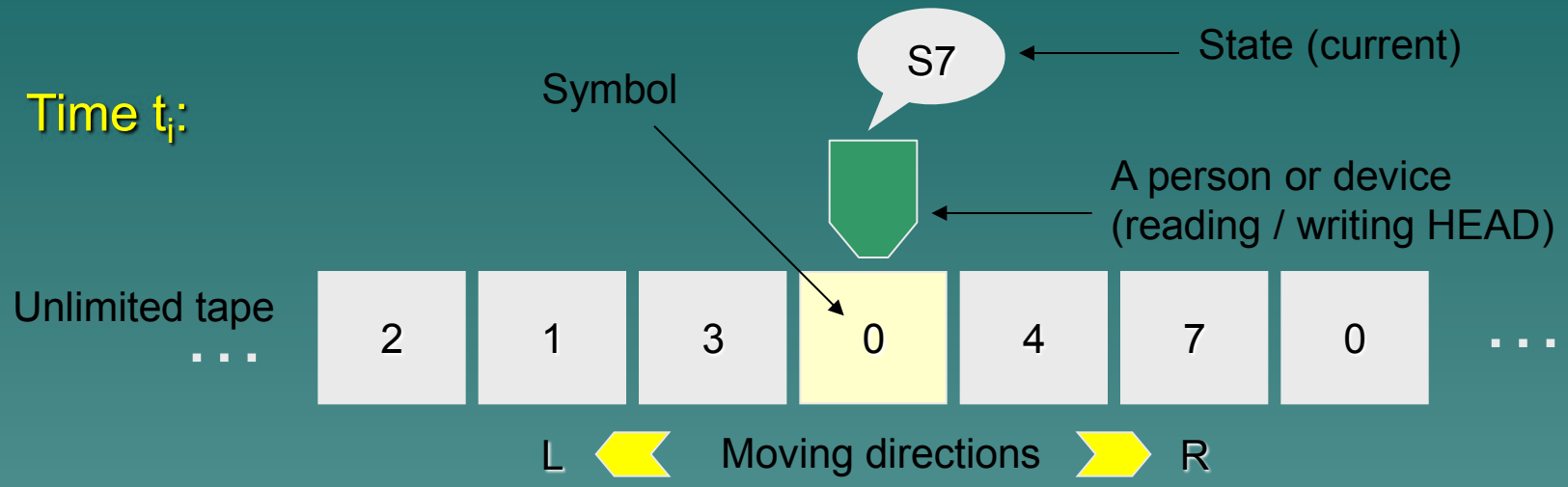
First concepts

Turing machines (1)

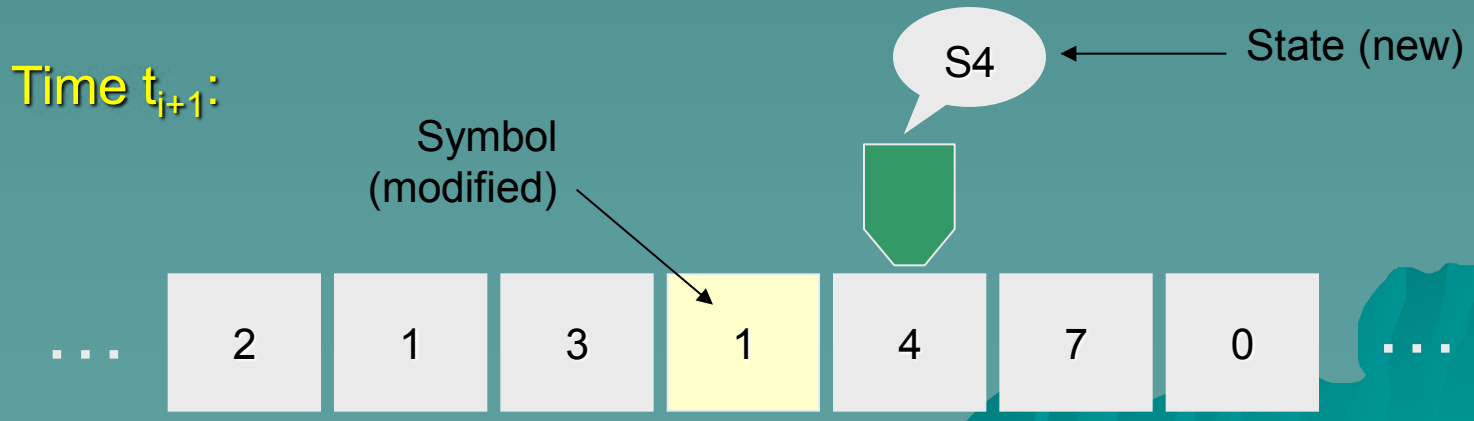
- Turing machines, described in 1936 by Alan Turing, are extremely basic abstract symbol-manipulating devices which can be adapted to simulate the logic of any computer that could possibly be constructed.
- They were not meant to be a practical computing technology, but a thought experiment about the limits of mechanical computation. Studying their abstract properties yields many insights into computer science and complexity theory.
- The concept is based on the idea of a person (or device) executing a well-defined procedure by changing the contents of an unlimited paper tape, which is divided into squares that can contain one of a finite set of symbols. The person needs to remember one of a finite set of states and the procedure is formulated in very basic steps in the form of "If your state is S7 and the symbol you see is '0' replace this symbol with '1', move the tape one step to the left, and assume state S4 as your new state."

First concepts

Turing machines (2)



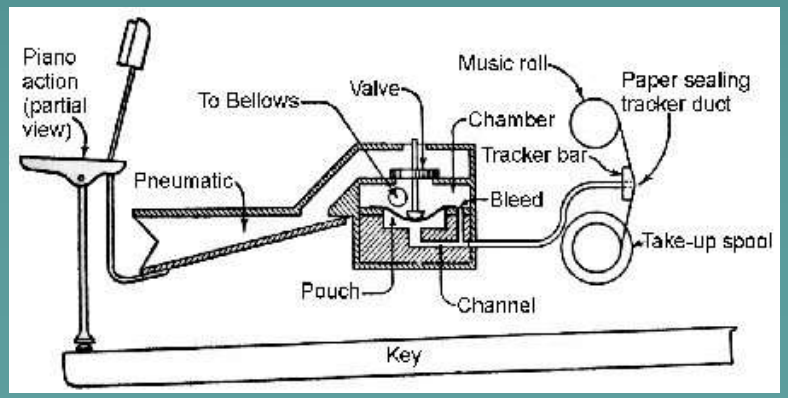
Instruction: "If your state is S7 and the symbol you see is '0' replace this symbol with '1', move the tape one step to the left, and assume state S4 as your new state."



”Mechanical ~Turing machines”

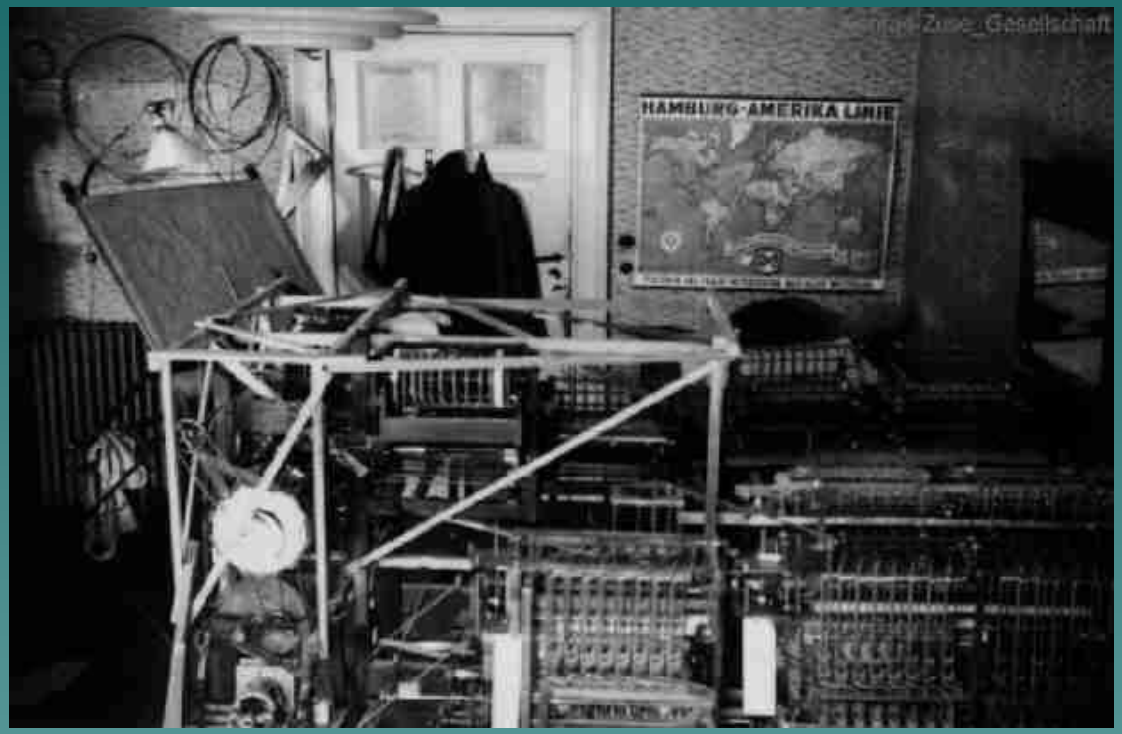
- Jacquard Loom is a automatic, mechanical loom, invented by Joseph Marie Jacquard in 1801.
- It utilized holes punched in pasteboard, each row of which corresponded to one row of the design. Multiple rows of holes were punched on each card and the many cards that comprised the entire design of the textile were strung together in order.
- The ability to change the pattern of the loom's weave by simply changing cards was an important conceptual precursor to the development of computer programming.

”Mechanical ~Turing machines”



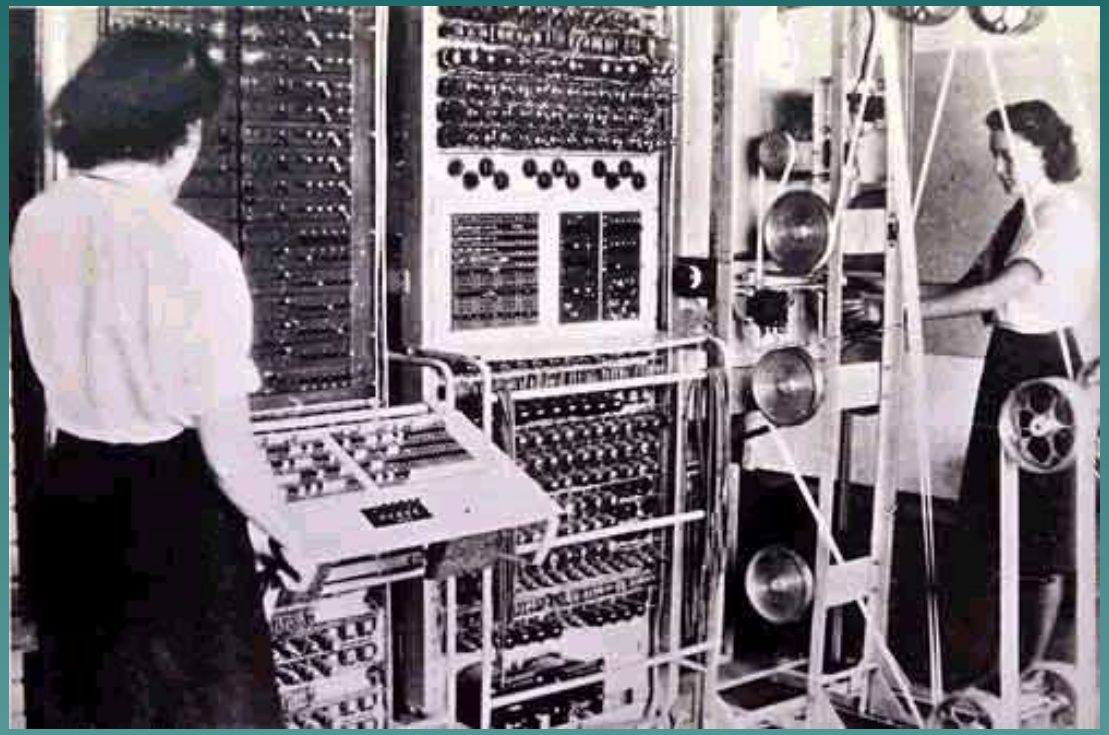
- Pianola – an example of automatic device for processing data of one kind: the music played on piano (first built in 1897 by E. S. Votey).
- The paper tape is limited, of course. The direction of traveling is only one: from the beginning to the end of the track.
- Each row on the paper tape includes set of holes, corresponding to piano keys. The mechanism opens or closes pneumatic valves driving the keys.
- The device can’t modify the tape itself during play. The recording (programming) is done ”off-line” with special pick-up.

First computers



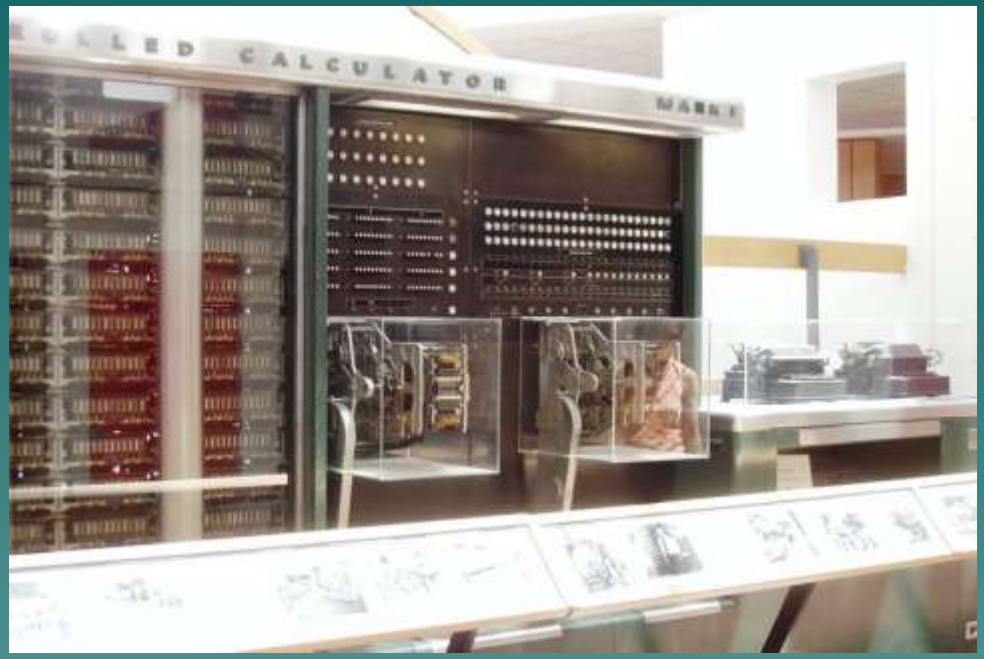
- Z1, Z2, Z3 (Konrad Zuse, 1941) – first ”generation 0” machines. Z3 was programmed by punched film tape, electromechanical calculator, compatible with Turing concept.

First computers



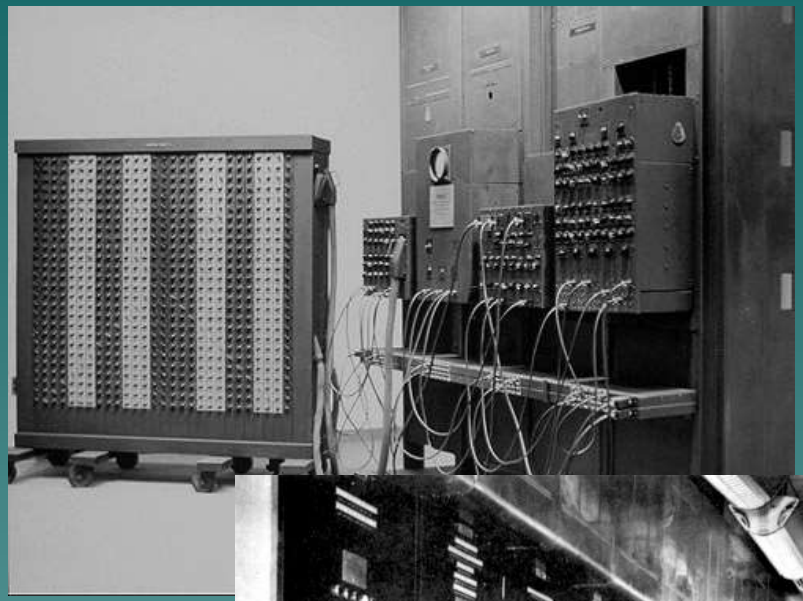
- Colosus (M. H. Newman, T. H. Flowers since 1943) – electronic programmable machines based on Turing concept (but with only one ”tape” traveling direction), programmed by rewireing. Used by British intelligence services during World War 2.

First computers



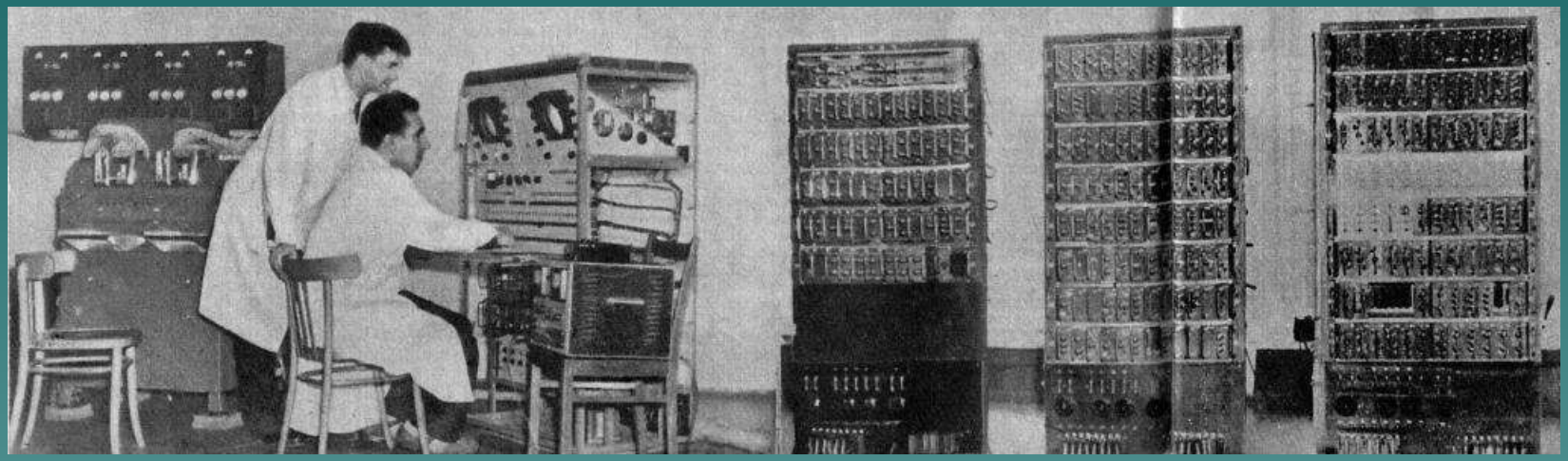
- IBM Harvard Mark I (Howard Aiken, 1944) – electromechanical Automatic Sequence Controlled Calculator (ASCC).
- It read its instructions from a 24 channel punched paper tape and executed the current instruction and then read in the next one. It had no conditional branch instruction. This meant that complex programs had to be physically long. A loop was accomplished by joining the end of the paper tape containing the program back to the beginning of the tape (literally creating a loop).
- Separately (not on the tape) it could store and modify 72 numbers, each 23 decimal digits long. This separation of data and instructions is known as the Harvard architecture.

First computers



- ENIAC (Eckert, Mauchly 1943 - 1945) – lack of RAM operational memory, programming by rewireing (later by function table in ROM).
- EDVAC (designed in 1952 by John von Neumann) – first computer which could load the program into the operational memory (1000 words in RAM) from the input device (mass storage), execute it and then reload or load another program. No physical separation of data and instructions. Input / output system based on punched cards and tapes from IBM.

Generation 1 computers



- XYZ (L. Łukaszewicz, 1958) – first computer designed and built in Poland. It's electronics included not only tubes, but also on semiconductors (Ge-based diodes).

Generation 2 and 3 computers

- 2-nd generation – discrete semiconductors (IBM 7094 – early 1960-ties).
- 3-rd generation – integrated circuits, low level of integration (IBM 360 – late 1960-ties).

- Multitasking operating systems.
- Modern programming languages.
- Many scientific and business applications.

Generation 4 computers (1)

- Minicomputers of the 1970-ties and early 1980-ties (PDP-8, PDP-11, VAX-11) was first ranked as 3-rd generation machines. In newer models however one LSI/VLSI chip (microprocessor) as CPU was introduced together with 16 or 32-bit architecture, first popular (not too expensive) multitasking operating systems like UNIX (K. Thompson, D. Ritchie) and it's clones, first TCP/IP networks and their services.
- Desktop (office) microcomputers of late 1970-ties and early 1980-ties – 8-bit architecture based on simple microprocessors like Intel 8080 (TRS-80), lack of multitasking, up to 64 kB RAM, CP/M operating systems (loaded from floppy-disks).

Generation 4 computers (2)

- Home microcomputers of early 1980-ties – 8-bit architecture based on simple microprocessors like Intel 8080, Zilog Z-80, Motorola 6800, Mostek 6510 (Sinclair ZX-Spectrum, Commodore C-64, Apple II etc.), lack of multitasking, up to 64 kB RAM, BASIC interpreter as operating system, cassette tape-recorder as mass-storage, TV-set as monitor.
- Millions of owners and fans, new market sectors (entertainment, education software etc.).

Generation 4 computers (3)

- Home microcomputers of late 1980-ties and early 1990-ties – 16 or 32-bit architecture based on microprocessors like Motorola 68000 (Atari ST, Commodore Amiga), multitasking operating systems, graphical user interface (GUI), good support for multimedia, floppies and HDD as mass-storage, high-resolution black & white or color CRT monitors.

Generation 4 computers (4)

- Desktop microcomputers of 1980-ties and early 1990-ties – 16 or 32-bit architecture based on microprocessors like Intel 80x86 (IBM-PC) or Motorola 68000 or newer (Apple Macintosh).
- In early IBM PC's (up to half of 1990-ties) very popular but not multitasking Disk Operating System (MS-DOS). Multitasking, virtual memory and GUI implemented initially as an extension (overlay) for MS-DOS in MS-Windows 3.0 and later Windows systems.
- In Macintosh computers multitasking and GUI from the very beginning. Less flexible hardware architecture (lack of extension slots in early models), much higher price.

Generation 4 computers (5)

- Today's laptop, desktop and enterprise microcomputers – 32 or 64-bit architecture based on Intel, AMD (marked as x86 and x86_64) and sometimes RISC processors of different types (professional workstations, enterprise class servers).
- Multitasking, virtual memory and GUI are implemented as a standard.
- Most popular operating systems are based on Windows-NT technology (Microsoft) or UNIX / Linux family of systems (Mac OS X, BSD, different Linux distributions etc.).
- Multi CPU mainboards and processors with 2 or more kernels are widely used for better multitasking and introducing parallel processing.

Generation 5 computers

- The Fifth Generation Computer Systems project (FGCS) was an initiative by Japan's Ministry of International Trade and Industry, begun in 1982, to create a "fifth generation computer" which was supposed to perform much calculation utilizing massive parallelism. The aim of the project was to create an "epoch-making computer" with supercomputer-like performance and usable artificial intelligence capabilities.
- Whereas previous computer generations had focused on increasing the number of logic elements in a single CPU, the fifth generation, it was widely believed at the time, would instead turn to massive numbers of CPUs for added performance.
- The project imagined a parallel processing computer running on top of massive databases (as opposed to a traditional filesystem) using a logic programming language to access the data.
- The FGCS Project did not meet with commercial success an was cancelled in 1993. The software was not suitable for commercial applications and the proprietary architecture was eventually surpassed in speed by less specialized hardware (Sun SPARC workstations and Intel x86 machines for example). The computers, operating system and programs produced by the project are now of historical, academic interest.

Generation 5 computers

- Although the Japanese FGCS project has failed, many of the ideas defined in the concept of the 5 generation computer are in use today. For example:
  - Computer systems (or supercomputers) built with large number of processors connecting with each other in one virtual machine (cluster) instead of computers with very sophisticated, huge CPUs.
  - Large databases serviced by queries in logic-oriented (in 1st approach) SQL language (SELECT <fields> FROM <TABLE> WHERE <logical condition> for example) are most important elements of today's professional applications.
  - In very large projects, like Electronic Health Record for example, data is not stored in one huge, central database but collected dynamically on demand from many local databases (different hospitals, clinics, insurance companies and other institutions). This is possible by means of standard, application level protocols for data interchange (like HL7 in medicine) and spread computing concept.
  - Spread computing (in LAN or WAN network environment) in general displaces centralized computing.
- In fact since half of 1990-ties we can say *"Network is the computer"* (John Gage, Sun Microsystems). That means, computer networks with large number of different services has became our information environment. The end-user interface to this environment in many cases is not even a PC – cellular phones, tablets and other mobile devices rather.

Homework:

- Download and install the ”i8080/Z80 Emulator” application (Windows installer is packed in a ZIP file).
- Download associated documentation (”Intel 8080 architecture description” document).
- Try to prepare and run the program from the ”Intel 8080 programming example 1” document according to explanation given in this document.