Operating Systems
Introduction

dr. Tomasz Jordan Kruk

T.Kruk@ia.pw.edu.pl

Institute of Control & Computation Engineering
Warsaw University of Technology

Operating Systems course (I)

- √ lecturer: dr. Tomasz J. Kruk, **T.Kruk@ia.pw.edu.pl**, **p. 530**
- √ course divided into lectures and laboratory, marked independently,
- √ grades: 40 points from laboratory and 30 points from the final test (or exam),
- √ necessary to achieve at least 20 points from the laboratory tasks to be admitted to the final test,
- √ laboratory tasks based on writing some programs in shell, C language and small items in Java language,
- √ each laboratory task is marked independently. The grade (number of points) is given by the person in charge of laboratory course and is final,
- √ laboratories are penalized for lateness,

Operating Systems course (II)

Laboratory tasks

1. 10 p. - shell programming
2. 5 p. - processes and signals
3. 5 p. - scheduling
4. 5 p. - memory management
5. 5 p. - synchronization
6. 5 p. - I/O operations
7. 5 p. - threads programming

Points from the lectures and laboratory are summed up. It is required to achieve at least 15 points from the final test and 35 points together. Grades:

0-34.5	2.0
35-41.5	3.0
42-48.5	3.5
49-55.5	4.0
56-62.5	4.5
63-70.0	5.0

It is possible in special circumstances to have the grade increased by the lecturer.

Operating Systems lectures

1. Introduction
2. Shell programming
3. Processes and threads
4. Interprocess synchronization and communication
5. Memory management
6. Input/Output system
7. File system
8. Multimedia operating systems
9. Multiprocessor systems and cluster solutions
10. Security

Operating System

Def. 1
Operating system is a set of programs and procedures performing two unrelated functions:

- √ management of resources by sharing (multiplexing) them in time and in space,
- √ presentation to the user the equivalent of an extended machine or virtual machine that is easier to program than the underlying hardware.

Def. 2
As an operating system **resource** each of the system's hardware and software element is considered iff that element may be allocated to some process.

Operating system resources

As *hardware resources* one can understand:

- √ processor time,
- √ memory,
- √ external devices,
- √ other computers available via network.

As *software resources* one can understand:

- √ files,
- √ buffers,
- √ semaphores,
- √ system tables.

Resources are managed in time and space.

Operating system resources management

Considering operating system as a resource manager it is in charge of:

- √ tracing the usage of system resources,
- √ forcing the strategy which contains and describes: a receiver, type of a resource, the moment of allocation and amount of allocated items,
- √ allocating resources,
- √ deallocating resources.

Virtual machine creation

As an **extended/ virtual machine creation** it is understood making to the user system abstraction easier to use and/ or program.
The goal is to present the user with the equivalent of an extended machine or virtual machine that is easier to program than the underlying hardware.

- √ transformation of the real machine into the machine which features required by assumed **processing mode** (destination of the computer system),
- √ exemplary abstractions:
  - ⋆ *„there is a group of named files stored on a disk"*,
  - ⋆ *„system enables concurrent execution of applications"*.

Processing modes

- √ **batch mode**, (off-line mode), autonomous usage of the computer without any need of presence of the user,
  - \+ huge throughput of the computer system,
  - – possible long time of waiting for task results, limited scheduling opportunities, inability of current control over execution process
- √ **interactive mode**, (on-line mode), conversational co-operation of the user with the system with usage of computer terminal,
  - \+ fast system response, ability of current control over execution process,
  - – less effective usage of computer system resources.
- √ **real time mode**, a system, which as a user has a technological process with some time constraints/ requirements. Two approaches:
  - ⋆ system has to react for external events/incidents faster than configured impassable time,
  - ⋆ system periodically monitors a state of the technological process.

The History of the Operating Systems

Charles Babbage (1792-1871) - project of the first true digital (but mechanical) computer. Ada Lovelace hired as the world's first programmer.

Generations of the operating systems:

1. The First Generation (1945-55) - vacuum tubes and plugboards.
2. The Second Generation (1955-65) - transistors and batch systems.
3. The Third Generation (1965-80) - integrated circuits and multiprogramming.
4. The Fourth Generation (1980-..) - personal computers.

The First Generation (1945-55) (I)

ENIAC - Electronic Numerical Integrator and Computer - built up at the University of Pennsylvania by J. Presper Eckert. and William Mauchley.

- √ built up in years 1943-1946, used till 1955,
- √ 30 tons, 1400 m2, 18 000 vacuum tubes, 140 kW,
- √ 5000 addition operations per second,
- √ 20 registers for ten digit decimal numbers,
- √ main drawback: programming via setting of relays (switches) and plugging and unplugging of cables.

The first generation characterized by lack of software.

The First Generation (1945-55) (II)

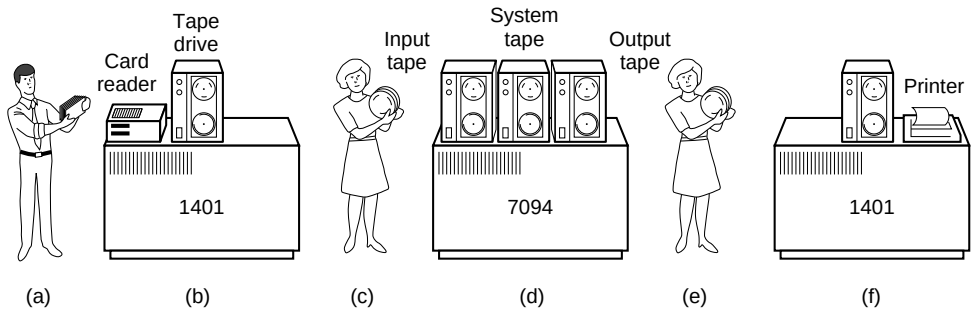
In the 1944 **John von Neuman** formulated the following assumptions:

1. Contemporary computer should contain:
   - √ the memory consisting of elements which may have only one of the following states: **0** or **1**,
   - √ **arithmetic unit**, which performs arithmetic, logic and other operations,
   - √ opportunity to input, output and control of data.
2. Activity of the computer is related to execution of some program and data, which are in the memory, processing. The program may contain *conditional operations*, which enable branches and jumps. The program may modify itself during the execution.

The Second Generation (1955-65)

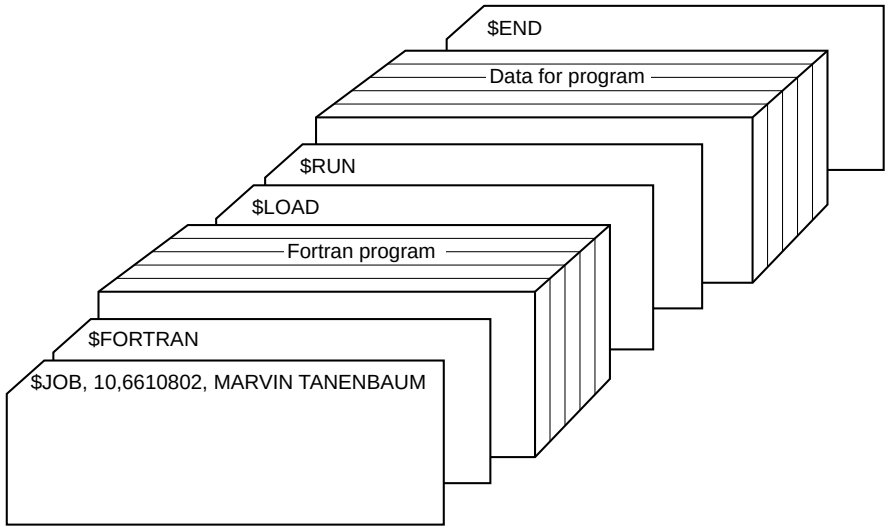
- √ transistors and batch systems,
- √ the epoch of so called **mainframes**,
- √ **batch processing** as a method of increasing the level of the utilization of the processor time,
- √ specialization of systems, exemplary assigns:
  - ⋆ **IBM1401** for input and outpu of data,
  - ⋆ **IBM7094** for numerical computations (number crunching).

Early Batch Systems



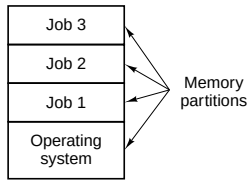
- √ FMS (ang. *the Fortran Monitor System*)

Structure of a typical FMS job



The Third Generation (1965-1980)

- √ usage of integrated circuits (ICs) for construction of IBM 360,
- √ **multiprogramming**,
- √ machines like IBM 1401 eliminated by **spooling** *Simultaneous Peripheral Operation On Line*,



- √ **timesharing** as a variant of multiprogramming, in which users could use at the same moment different terminals,
- √ **Compatible Time Sharing System** (CTSS) as a first serious time sharing system (M.I.T., 1962),
- √ batch processing and interactive processing.

Systems of the Third Generation

- **MULTICS** (ang. *MULTiplexed Information and Computing Service*) - MIT, Bell Labs, General Electric - project of a single system which had to be capable of servicing with computational power all the region of Boston (*www.multicians.org*),
- **DEC PDP** PDP-1 (1961 r.) started the era of minicomputers - almost as fast as IBM 7094, but for the price of 5% of 7094's; the success story of the PDP family, especially **PDP-11**,
- **Ken Thompson** from Bell Labs on his PDP-7 created limited one-person version of the Multics, which evolved into **Unix** system,
- two main branches/ lines of the Unix system: **System V** (AT&T) and **BSD**,
- **POSIX** (ang. *Portable Operating System Interface*) - defined by IEEE standard of conformance with Unix system (now: with open system).

The Fourth Generation (1980-today) (I)

- personal computers (*microcomputers*) similar in architecture to minicomputers but definitely different in price,
- 1974, **Intel** introduces **8080**, first general purpose 8-bit processor, Gary Kildall writes **CP/M** (Control Program for Microcomputers),
- 1977, CP/M rewritten by Digital Research for other microprocessors, 5 years of domination of CP/M,
- the beginning of eighties - IBM developed IBM PC and contacted **Bill Gates** for licensing BASIC language interpreter,
- the worst decision of Digital Research - lack of commitment in talks with IBM,
- B. Gates bought from Seattle Computer Products operating system **DOS** and offered tandem DOS/BASIC for IBM,

The Fourth Generation (1980-today) (II)

- **MS DOS** (ang. *MicroSoft Disk Operating System*), the new version of DOS rewritten by hired by Microsoft creator of original DOS,
- **GUI** (ang. *Graphical User Interface*) invented by Doug Engelbart in Stanford Research Institute in sixties; adopted by Xerox PARC scientists,
- **Steve Jobs**, Apple co-creator - an attempt to build an Apple with GUI, systems **Lisa** and **Apple Macintosh**,
- systems **Windows 3.11**, Windows 95, Windows 98, Windows ME,
- system **Windows NT** developed in huge percentage by again hired by Microsoft developers of VAX VMS system.
- clone of the Unix system, **Minix**, written by **A. Tanenbaum**,
- a new system, with roots in the Minix, **Linux**, created by **L. Torvalds**.
- the time of network and network operating systems,

Different types of operating systems

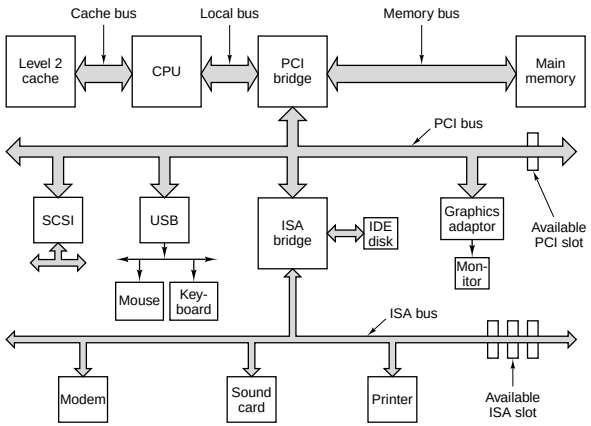
There are different types of operating systems, although they have much in common. There are operating systems:

- for *mainframes*,
- for servers,
- for multiprocessor machines,
- for personal computers,
- real-time operating systems, with hard and soft limits (*VxWorks*, *QNX*),
- dedicated for embedded systems (*PalmOS*, *Windows CE*),

Review of computer architectures (I)

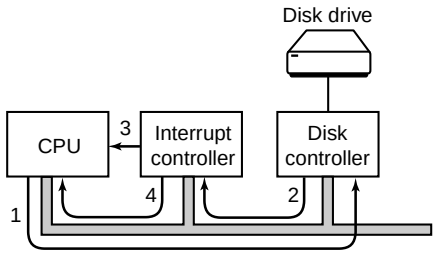
- √ processor: program counter, stack pointer, PSW,
- √ memory: RAM, ROM, EEPROM, flash RAM, physical and virtual addresses, MMU,
- √ different buses:
  - ⋆ IDE (ang. *Integrated Drive Electronics*),
  - ⋆ ISA (ang. *Industry Standard Architecture*),
  - ⋆ PCI (ang. *Peripheral Component Interconnect*),
  - ⋆ USB (ang. *Universal Serial Bus*),
  - ⋆ SCSI (ang. *Small Computer System Interface*),
  - ⋆ IEEE 1394 (FireWire)

Review of computer architectures (II)

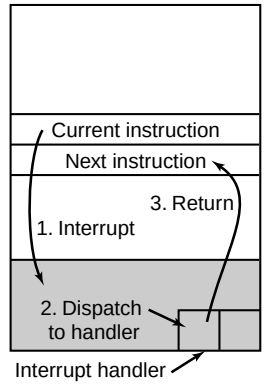


Structure of the computer system with the Pentium-like architecture.

Interrupt handling in the computer system



(a)



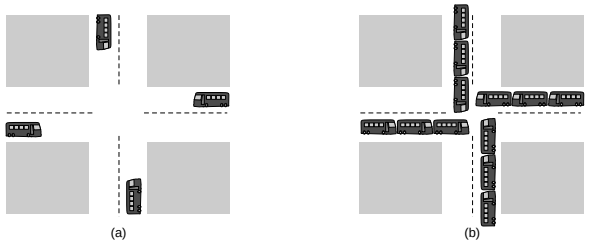
(b)

√ Interrupt handling in the computer system.

Concepts related to operating systems

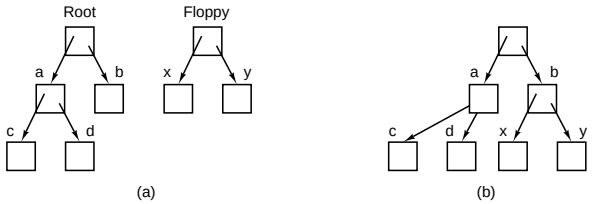
- √ process, address space of a process, table of processes, process image, child processes, interprocess communication, signals, identifiers: of process, of group, of process' owner,
- √ deadlocks,
- √ memory management,
- √ input/ output,
- √ files, directories, access paths, root directory, current directory, file descriptor, file system, special files, block and character devices, pipes,
- √ security, **rwx** bits, access control lists,
- √ command interpreter, shell.

Deadlocks in operating systems



a. potential deadlock,
b. real deadlock.

External file system mounting



a. before mounting - data from the floppy unavailable,
b. after mounting - data on the floppy available.