1.8.2008 / Teemu Kerola

Learning Goals for Computer Organization I

Main theme	***Prerequisites***	***Approaching Learning Goals***	***Reaches Learning Goals***	***Deepens Learning Goals***
Computer system structure and operation. *Can explain main system components and how they work for one program in execution.*	Can explain various computer parts from user stand point (Introduction to the Use of Computers).	Can explain processor operation (fetch–execute cycle) in executing machine instructions. Can explain basic ideas in bus and memory hierarchy. Can explain different CPU execution modes and especially the meaning for privileged execution mode. Can explain the operation of the interrupt sybsystem.	Can explain large system component speed differences and their effect on the system. Can explain the fundamentals of cache and virtual memory. Can explain when and how processor execution mode changes. Can explain the significance of interrupts and their implementation at machine instruction level.	Can explain, how machine instructions are executed at clock pulse level. Knows the history of memory technology development and various technologies used for it. Can explain the implementation of multi-level virtual memory.
Machine language programming and program execution. *Can implement basic programming concepts with symbolic assembly language designed for education.*	Can program simple programs with some block structures language (Introduction to Programming). Can use (unsigned) integers, rational and real numbers in various applications (Introduction to Programming).	Can use global variables, constants, records and simple arrays, as well as explain their implementation during compilation, loading and execution. Can program selection and loop structures as well as implement subroutine calls. Can explain integer and floating point data representation formats as well as basic ideas of character and string data formats. Can transform integer data representations between 2, 8, 10 and 16 based representations.	Can implement data storage of records, objects and multi-dimensional arrays (of different types) and references to them. Can explain differences between global and local data structures in data allocation and references, and can use activation records and activation record stack for (recursive) subroutine implementation. Can explain the basics of (moving) picture and sound formats, and can explain and take into consideration floating point data accuracy changes during computations. Can use basic methods for data integrity protection (parity bit, Hamming code) .	Can explain precisely Java objects implementation at machine language level Can write execution time optimized code Can explain how CRC is used for data integrity Can explain the reasoning for redundant devices in securing data access and data integrity.

Main theme	***Prerequisites***	***Approaching Learning Goals***	***Reaches Learning Goals***	***Deepens Learning Goals***
Operating system (OS) basic structure and operation. *Can explain OS basic structure, services and functionality from the viewpoint of one program in execution. Can explain different methods to execute programs.*	Can write simple programs using high level programming language (Introduction to Programming) Can use simple OS services in programs as well as within OS user interfaces (e.g., open file, execute program).	Can explain the process concept and its representation in operating system as well as various process states and transitions between them. Can list fundamental OS functionalities (process, memory, file, device and network management). Can explain basic ideas in compilation, linking and loading. Can explain the structure and operation of disk drives as well as compute access time for data stored in hard drive. Can classify I/O implementation methods. Can explain various methods to execute Java programs, and differences between them.	Can explain when and how the process in execution changes. Can explain OS goals and basic functionality both from user and system manager viewpoint. Can explain OS implementation with (privileged) subroutines and processes. Can explain the (dis)advantages of static and dynamic linking. Can explain how OS device interface is implemented with device drivers and interrupt subsystem. Can explain the main idea of file server and file cache. Can explain the basic idea and operation of Java Virtual Machine (JVM), and can select the correct Java execution method for given application. Can explain the basic idea in JIT-compilation.	Can explain details of given dynamic linking methods. Can explain how OS services are invoked with various (privileged) call mechanisms. Can explain the exact structure of the JVM or all instructions in Java byte code. Can explain, how complete processor operation may be based on emulation, or how ordinary machine language programs can be executed in any machine architecture with emulation.
(Web-based learning methods)			(Can describe the differences between Authorware-type web lectures, ordinary lectures and text books.)	(Can use discussion groups, blogs and wikis for team work and learning.)