



Univerza v Ljubljani
Fakulteta za računalništvo
in informatiko

First cycle university study programme, 3rd year

System Software

lecturer: assist. prof. Tomaž Dobravec

Introduction

About the course

- ▶ Lecturer: assist. prof. **Tomaž Dobravec** (tomaz.dobravec@fri.uni-lj.si)
- ▶ Timetable: **Thursday at 11.15 – 13.45 (P19)**
 - One break (12.30-12.45)
- ▶ Office hours: **Tuesday at 1 p.m.**
- ▶ Contact page: eUčilnica

About tutorials

- ▶ Assistant: assist. prof. Jurij Mihelič (jurij.mihelic@fri.uni-lj.si)
- ▶ Start on October 12th, 2020

Details about tutorials will be given by Jurij Mihelič

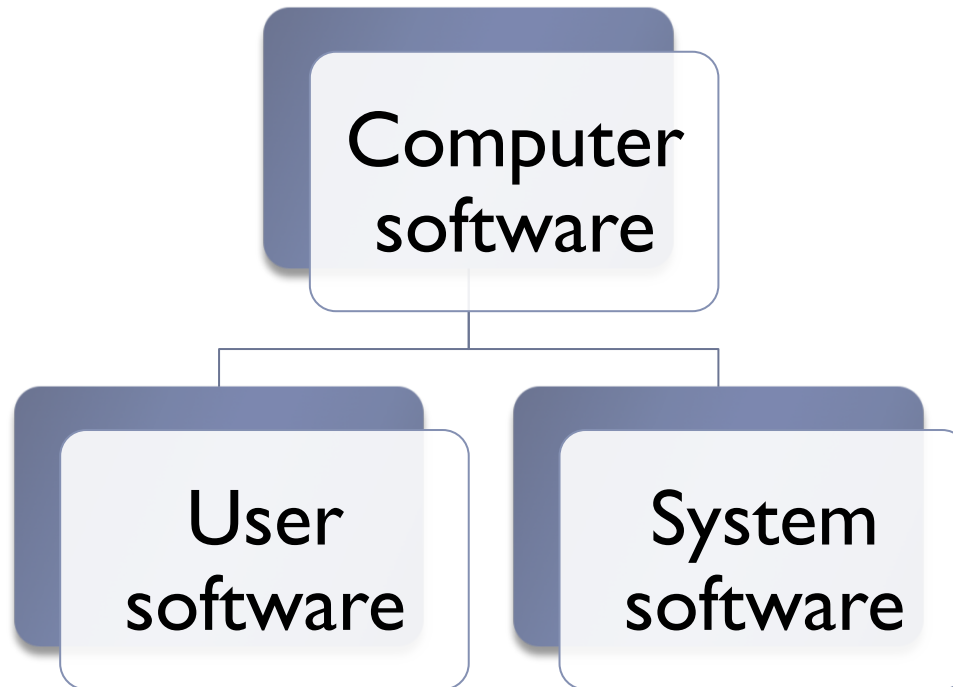
Literature

- ▶ Main: Leland L. Beck: System Software (An introduction to System s Programming), Addison Wesley, 3th edition, 1997
- ▶ Additional: Damjan Zazula, Mitja Lenič: Principi sistemske programske opreme, Univerza v Mariboru, FERl 2006
- ▶ See also:
 - ▶ Source code on eUčilnica ([Lecture source files](#))
 - ▶ The web sources listed on eUčilnica ([Resources](#))

Requirements

- ▶ Assessment = tutorials (50 points) + written exam (50 points)
- ▶ Prerequisite for taking the written exam:
at least 25 points (50%) from the tutorials.
- ▶ To pass the exam:
at least 25 points (50%) on written exam.
- ▶ Tutorials: Seminar (20 points) + homework (30 points).
(for details about homework – see Tutorials Introduction on eUčilnica)

Computer software is roughly divided into two groups



User software includes programs that make the operating system user-friendly.

- ▶ Programs for working with text and spreadsheets,
- ▶ Web browsers,
- ▶ Systems for working with databases,
- ▶ Programs for performing various analyzes,
- ▶ Drawing and planning programs,
- ▶ ...

System software includes programs, standards and principles for software development and for system maintenance

- ▶ **Development tools** (compiler, assembler, interpreter, linker, loader, debugger, patcher ...)
- ▶ Tools for working with file system and network,
- ▶ System status information (time, amount of free memory, information on logged-in users)
- ▶ Communication mechanisms (between processes, programs, users and various computer systems)
- ▶ System files (libraries and services, device drivers, system settings, configuration files, ...),
- ▶ Tools for working with text files (grep, sed, awk, sort, join, ...),
- ▶ ...

- ▶ System software is **highly dependent** on the computer system.
- ▶ Nevertheless: a design and the operational logic is the same (or at least very similar) for all the systems.
- ▶ System Software: background, low-level programs for experienced users and developers

Goals of the System Software Course

1. To learn the logic and the design of the development tools: assembler, linker, loader, ... (the first part of the lectures and most of the tutorials)
2. To learn a wide range of topics covered by the System Software (seminars)

Seminars

- ▶ Choose and examine a topic, prepare a presentation (as a PDF document) and present a topic in the classroom
- ▶ The content of the seminars is a part of the learning material.
- ▶ Details: in the following

Seminars

Each student prepares a seminar in one of the areas listed below. The purpose of each seminar is:

- ▶ to show a system-independent logic of the selected area,
- ▶ to show its specifics for popular operating systems (Windows, Linux, MacOS, Android, ...),
- ▶ to show the implementation of the presented concepts in selected operating systems,
- ▶ to provide (at least a simplified version of) a tool covering the presented area.

Seminars

- ▶ When presenting the topic and tools, **WE ARE NOT INTERESTED** in the user experience
- ▶ Instead: we are interested in the **BACKGROUND** of the whole story (how the things really work, which algorithms and what standards are used ...)

Development tools

- ▶ Debugger
- ▶ Patcher
- ▶ automatization tools (Make, Ant, autoconfig)

System tools

- ▶ **System information:** about hardware, software, system status (time, amount of free memory, processor load, network, logged-in users ...)
- ▶ **Security** (data encryption tools, digital signature, checksum, ...)
- ▶ **Screen saver**
- ▶ **Tools for automatic or manual updating** of system software in different operating systems.
- ▶ **BenchMarking** software
- ▶ **Clipboard**
- ▶ High-level tools for working with **text files**: awk, groff, LaTeX, ..

System files

- ▶ **Registry** in Windows: aim, standard and tools. Equivalent (or alternative) in the other operating systems.

- ▶ Log files: aim, standard and tools.

Hard drive management tools

- ▶ **Defragmentation:** searching for files that are "fragmented" over the disk and the ways to rearrange (aggregate) files.
- ▶ **Disk check:** searching for corrupted or incorrectly stored files, searching for defected areas of the disk, searching for unused and large files, finding ways to restore data.

Data management tools

- ▶ Disk and files **backup**
- ▶ **Storing and archiving data** (zip, rar, ...).

Virtualization

- ▶ **Virtual machines** (VMWare, Xen, ...)
- ▶ **Sandboxes** (sandboxes)

- ▶ Any other topic in the field of System Software (student's choice)
 - ▶ be aware: the student should consult a professor or assistant on the appropriateness of the chosen topic.