Linux workshop

Syllabus

Requisites of the Course	
Cycle of Higher Education	**First cycle of higher education (Bachelor's degree)**
Field of Study	12 Information Technologies
Specialty	121 Software engineering
Education Program	Computer Systems Software Engineering
Type of Course	Normative
Mode of Studies	full-time
Year of studies, semester	Second year, first semester
ECTS workload	6.5 credits (ECTS). Time allotment - 195 hours, including 72 hours of classroom work, and 123 hours of self-study.
Testing and assessment	Exam
Course Schedule	2 classes per week by the timetable http://rozklad.kpi.ua/
Language of Instruction	English
Course Instructors	Lecturer and lab teacher: PhD, Associate Professor, Oleksandr Rokovyi, email: rokovyi@comsys.kpi.ua
Access to the course	https://cloud.comsys.kpi.ua/s/RLQxx7rcCEYmAAB

Outline of the Course

1. Course description, goals, objectives, and learning outcomes

You can effectively study modern information technology only with the use of open source software. Access to the source code has a positive effect on the learning process, as it allows you to understand how the system works from the inside. The future specialist must have a deep knowledge of software and hardware of computer. To do this, you should use an operating system that provides such capabilities. One example of such systems is GNU / Linux, which is a free and open source software.

The GNU / Linux operating system, with open source code, can be easily ported to a variety of hardware platforms. This makes it one of the most common operating systems in the world. This system can be used on a personal computer, laptop, server, mobile device, embedded system. That is why the future specialist must at least have the skills to work with this system.

The GNU / Linux operating system is an effective tool used in various areas of information technology: computer networks and services, server platforms, cybersecurity, embedded devices, the Internet of Things.

The purpose of the course is to train specialists who have knowledge of the architecture and principles of building the GNU / Linux operating system, as well as practical skills in using this system as a tool for solving various tasks.

The subject of the discipline - theoretical and practical foundations of construction and operation of operating systems on the example of GNU / Linux.

Discipline "Linux workshop" provides the following program competencies and program results of educational and professional program: FC03, FC15, PRN25, PRN26:

- the ability to develop architectures, modules and components of software systems;

- the ability to develop and use network technologies;

- know the software of high-performance systems;

- know the principles of construction and operation of high-performance computer systems.

According to the curriculum of the discipline, students after mastering the discipline must demonstrate the following program learning outcomes.

Knowledge:

- basics of operating system architecture on the example of GNU / Linux operating system;

- functions of operating system components and their interaction;

- GNU / Linux file system hierarchy standard;

- main types of files and their purpose;

- life cycle of processes and their types;

- basic operators for building scripts.

Skills:

- install the GNU / Linux operating system on any hardware platform;

- be able to install new, update existing and remove unnecessary applications;

- manage user and group accounts in the GNU / Linux operating system;

- set restrictions on access to files and directories;

- configure equipment;

- configure the parameters of network interfaces;

- mark media and create file systems on them.

Experience:

- work with the main commands of the GNU / Linux operating system;

- construction of chains of commands of arbitrary length to solve problems;

- writing scripts in the built-in shell programming language;

- filtering text streams.

2. Prerequisites and post-requisites of the course (the place of the course in the scheme of studies in accordance with curriculum)

In order to successfully master the discipline "Linux workshop" in accordance with the educational program, it is necessary to master the knowledge of the discipline: "Basics of computer systems and networks", "Basics of programming".

Competences, knowledge and skills acquired in the study of the discipline "Linux workshop" can be used to study the disciplines: "Networks and network information technologies", "Operating systems", "Software Security".

3. Content of the course

Chapter 1. Introduction to GNU / Linux

Topic 1.1. Basics of the GNU / Linux operating system.

Topic 1.2. Features of the GNU / Linux installation process.

Chapter 2. Learning GNU / Linux command-line tools.

Topic 2.1. Work in the command line.

Chapter 9. X Window System Setup, Localization, and Printing.

Topic 9.1. Installing and configuring X11.

Topic 9.2. Display manager settings.

Topic 9.3. Special opportunities.

Topic 9.4. Localization and internationalization.

Topic 9.5. Manage printers and print.

4. Coursebooks and teaching resources

Main literature.

1. William Shotts. The Linux Command Line. Fifth Internet Edition, 2019. – 555 p. URL: https://sourceforge.net/projects/linuxcommand/files/TLCL/19.01/TLCL-19.01.pdf/download

2. Brian Ward. How Linux Works, 3rd Edition: What Every Superuser Should Know. No Starch Press, Inc. 2021. – 467 p.

3. Christopher Negus. Linux Bible, Tenth Edition. John Wiley & Sons. 2020. – 928 p.

Supplementary literature.

1. Andrew Tanenbaum, Herbert Bos. Modern Operating Systems, 4th Edition. Pearson Higher Education. 2015. – 1137 p.

2. William Stallings. Operating Systems: Internals and Design Principles, 9th Edition. Pearson Education, 2018. – 1128 p.

3. Paul Cobbaut. Linux Fundamentals, 2015. – 365 p. URL: http://linux-training.be/linuxfun.pdf

4. Paul Cobbaut. Linux System Administration, 2015. – 385 p. URL: http://linux-training.be/linuxsys.pdf

5. Paul Cobbaut. Linux Security, 2015. – 129 p. URL: http://linux-training.be/linuxsec.pdf

6. Paul Cobbaut. Linux Networking, 2015. – 294 p. URL: http://linux-training.be/linuxnet.pdf

7. Paul Cobbaut. Linux Storage, 2015. – 278 p. URL: http://linux-training.be/linuxsto.pdf

8. Introduction to Linux for Users and Administrators. Version 4, 2015 – 176 p. URL: https://www.tuxcademy.org/download/en/grd1/grd1-en-manual.pdf

9. Linux Administration I System and Users. Version 4, 2015 – 238 p. URL: https://www.tuxcademy.org/download/en/adm1/adm1-en-manual.pdf

10. Linux Administration II Linux as a Network Client. Version 4, 2015 – 217 p. URL: https://www.tuxcademy.org/download/en/adm2/adm2-en-manual.pdf

Educational content

5. Methodology

5.1. Lectures

№	Lecture topic and list of main issues	Number of hours
1	**Basics of the GNU / Linux operating system.** Course structure. Purpose and functions of GNU / Linux. History of GNU / Linux. GNU / Linux distributions. Basic ways to use GNU / Linux. GNU / Linux installation process.	2

№	Lecture topic and list of main issues	Number of hours
	Self-study: Get acquainted with the features of distributions: Ubuntu, Debian, Red Hat, CentOS, Fedora, OpenSUSE.	
2	**Learn GNU / Linux command line tools.** GNU / Linux operating system installation. Work in the command line. Bash shell. Environment variables. Use help on operating system commands and components. Commands history. Absolute and relative path. File system tree navigation. **Self-study**: Get acquainted with advanced bash shell settings.	2
3	**Processing text streams using filters.** Use GNU / Linux command line filtering commands. **Self-study**: Get acquainted with the language of processing templates awk.	2
4	**Standard I / O streams.** Use of streams, pipelines and redirects. Command control operators. Use special command line characters. **Self-study**: Ability to use tee and xargs commands.	2
5	**Regular expressions.** Use regular expressions. Symbols, operators and metacharacters in regular expressions. Character classes. **Self-study**: Ability to use the grep command.	2
6	**Basics of file and directory management.** File types in GNU / Linux. Create, copy, move and delete files. Use templates to work with files on the command line. Hard and symbolic links. **Self-study**: Get acquainted with the structure of index descriptors.	2
7	**File permissions.** Security model in the GNU / Linux operating system. Permissions and ownership in GNU / Linux. Change the file access mode. Change file owner. Advanced file access attributes. **Self-study**: Familiarize yourself with access control lists (ACLs) in GNU / Linux.	2
8	**File system hierarchy standard.** File System Hierarchy Standard (FHS). Search for files in the file system tree. **Self-study**: Get acquainted with the structure of virtual file systems procfs and sysfs.	2
9	**System administration.** Manage users and groups, and related system files. The structure of files that store system account information. Logging operations in GNU / Linux. Task Scheduler in GNU / Linux. **Self-study**: Get acquainted with the mechanism for setting quotas for the use of storage space. Set quotas for existing users and check their performance.	2
10	**Process management in GNU / Linux.**	2

№	Lecture topic and list of main issues	Number of hours
	Interactive and background processes. Kill command signals. Screen terminal emulator manager. Monitoring and killing of processes. Changing the priority of processes. **Self-study**: Get acquainted with the mechanisms of process control through the virtual file system procfs.	
11	**Software management.** Using the Debian Package Manager. Using Red Hat package managers. Managing shared libraries. **Self-study**: Create your own installation package for GNU / Linux.	2
12	**Hardware management.** MBR and GPT drive structure. Logical Volume Manager (LVM). Creating partitions and file systems. **Self-study**: Increase the size of the LVM volume and the file system in it without unmouting it from the file system tree.	2
13	**Working with file systems.** File system integrity support. File system mount management. **Self-study**: Get acquainted with the GParted utility, resize an existing Linux partition, change the file system to XFS.	2
14	**Configure network settings.** Basic concepts of Internet protocols. Basic configuration of network parameters. DNS client settings. **Self-study**: Create alternate entries in the routing table that do not change the logic of the software router.	2
15	**Network troubleshooting.** Basic approaches to solving network problems. TCP / IP diagnostic stack utilities. **Self-study**: Add the Wireshark utility to Linux with a GUI installed. Get acquainted with the main features of the Wireshark sniffer.	2
16	**Booting the GNU / Linux operating system.** The process of booting the operating system. Change system startup, shutdown, or reboot levels. System initialization in SysV style. System initialization in systemd style. Operating system boot managers. **Self-study**: Change the default operating system startup level to single-use.	2
17	**Writing scripts in bash.** Basic bash operators. Cycles. Conditions. Debugging scripts in bash. **Self-study**: Create a script that includes all types of loops, input and output operations.	2
18	**Operating system security.** Performing security management tasks. Host security settings. Data protection with encryption. **Self-study**: Familiarize yourself with the SELinux security system.	2
	Total:	36

5.2. Laboratory classes (computer workshop).

The main task of the cycle of laboratory classes (computer workshop) is to provide students with the necessary practical skills in the GNU / Linux operating system, the use of basic tools and the built-in documentation system.

№	Topic of laboratory work (computer workshop)	Number of hours
1	**Installing the GNU / Linux operating system.** Familiarity with the command line interface, the use of environment variables, getting help on commands.	2
2	**Familiarity with the GNU / Linux command line.** Environment variables, system manual pages.	2
3	**Processing text files using filtering commands.** Familiarity with the commands: cat, cut, tail, less, head, join, paste, nl, grep, sed, toilet, sort, uniq, split.	3
4	**Standard I / O streams, pipelines, and redirects.** Familiarity with commands: tee, xargs.	3
5	**Working with files, searching for files.** Familiarity with the commands: mkdir, rmdir, rm, mv, cp, ls, find, touch, file, dd. Use templates to work with files.	3
6	**Process control.** Familiarity with process control commands: top, ps, pstree, kill, pgrep, pkill, nice, renice.	2
7	**Users and permissions.** Familiarity with commands for managing user accounts: useradd, userdel, usermod, passwd, commands for changing file permissions: chown, chgrp, chmod.	3
	Total:	18

To successfully master the discipline, each student must prepare a workplace in the form of a virtual machine. The use of virtualization technology will avoid the problems associated with the restoration of the operating system in the event of errors during laboratory work. It will also allow you to use any operating system (not GNU / Linux) in the workplace. It is recommended to use the VirtualBox virtualization system (https://www.virtualbox.org/wiki/Downloads)

It is recommended to use Ubuntu (https://ubuntu.com/download/alternative-downloads) as one of the latest stable versions for laboratory work as a training distribution. To reduce the resource requirements for the hardware platform, you can use the option of installing the operating system without a graphical user interface.

To fully perform all laboratory work, each virtual machine must have an Internet connection.

6. Self-study

6.1. Topics that are submitted for self-study.

№	Topic submitted for self-study	Number of hours
1	Setting up the X Window System.	7
2	Setting up Display manager.	6
3	Printer and print management.	5
4	Localization and internationalization.	4
5	Linux kernel module management.	5
6	Data archiving tools in the GNU / Linux operating system.	4
7	Text editor vi.	4
	Total:	35

Before each class, the student performs self-study training in accordance with the topic of the lecture or laboratory work for at least two hours. Preparation for the exam must be at least 16 hours.

Thus, the student's self-study work during the semester should be: 36 + 36 + 35 + 16 = 123 hours.

7. Course policy

Course policy:
- for successful study of the discipline it is desirable to be present at all lectures;
- in lectures it is allowed to use any technique only for the purpose related to the lesson, without disturbing other students and teachers;
- during the lecture you can ask questions to the teacher, for this you need to raise your hand and get permission;
- it is forbidden to speak at lectures without the permission of the teacher;
- in lectures it is forbidden to engage in activities that do not directly relate to the discipline;
- laboratory work takes place in the form of a computer workshop;
- only students who are ready to defend their work should be present at laboratory classes;
- during the defense of the laboratory work the student must demonstrate the completed task and answer the teacher's questions (questions from the theory, practical task, etc.);
- options (if the division into options is provided in the task) for laboratory work are selected as follows: the first 15 students receive options according to the number in the group list, the student with number 16 in the list receives option 1, etc.
- penalty points for late protection of laboratory work are not accrued;
- it is possible to defend laboratory works in any sequence;
- repeated protection of laboratory works is prohibited;
- it is forbidden to use outside help during the defense of laboratory work.

8. Monitoring and grading policy

The final rating of a student in the discipline "Linux workshop" consists of points that he receives:
- for academic work during the semester (starting scores);
- for exam.

8.1. Getting of starting scores.
During the semester the student performs 7 laboratory works.
For each laboratory work the student receives:
- 8 scores for completed in full and without significant errors task for laboratory work;

- from 0 to 4 scores for the defense of laboratory work, which consists of theoretical questions, practical tasks.

1 modular test is planned, for which the student can receive from 0 to 16 points.

The student's starting scores are calculated as the sum of scores for all laboratory works and modular test multiplied by a coefficient of 0.6.

8.2. Conditions for admission to the final test.

To obtain admission to the final test student must defend 7 laboratory works.

8.3. Scoring for the exam.

In the exam, students must answer three theoretical questions. Each theoretical question is evaluated in 20 score.

Theoretical evaluation system:

18-20 score - complete answer (not less than 90% of the required information);

15-17 score - a fairly complete answer (at least 75% of the required information, or minor inaccuracies);

12-14 score - incomplete answer (at least 60% of the required information and some errors);

0 score - unsatisfactory answer (less than 60% of the required information or significant errors).

The sum of starting score and score for the exam are adopted by university grading system as follows:

Score	Grade
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactory
64-60	Sufficient
Below 60	Fail
Course requirements are not met	Not Graded

9. Additional information about the course

In addition to presentations, you should use a GNU / Linux command line terminal to demonstrate live examples. This will allow the student to learn more about the lecture material, as well as get used to the command line interface.

9.1. List of theoretical questions for the final test.

• Purpose and functions of GNU / Linux.

• Bash shell.

• Environment variables in bash.

• Absolute and relative way.

• Team history in bash.

• Standard I / O streams.

• Construction of chains of commands by means of the conveyor.

• Command control operators.

- Command line special characters.

- Use of regular expressions.

- Symbols, operators and metacharacters in regular expressions.

- File types in GNU / Linux.

- Use templates to work with files on the command line.

- Hard and symbolic links.

- Security model in the GNU / Linux operating system.

- Access rights and file ownership mechanism in GNU / Linux.

- Change the file access mode.

- Advanced file access attributes.

- File System Hierarchy Standard (FHS).

- Search for files in the file system tree.

- Manage users and groups, and related system files.

- The structure of files that store system account information.

- Logging operations in GNU / Linux.

- Task scheduler in GNU / Linux.

- Interactive and background operation of processes.

- Kill command signals.

- Process priorities.

- Debian Package Manager.

- Red Hat Package Manager.

- Management of shared libraries.

- Structure of MBR and GPT drives.

- Logical Volume Manager (LVM).

- Management of mounting file systems.

- Basic concepts of Internet protocols.

- Basic configuration of network parameters.

- DNS client settings.

- Basic approaches to solving network problems.

- TCP / IP protocol stack diagnostic utilities.

- The process of booting the GNU / Linux operating system.

- Change system startup, shutdown, or reboot levels.

- SysV-style system initialization.

- Systemd-style system initialization.

- GRUB2 operating system boot manager.

- Basic bash operators, organization of cycles, branches.

- Debugging scripts in bash.

- Security management tools in the GNU / Linux operating system.

- Data protection with encryption.

- SELinux security system.

9.2. Additional Information Resources

9.2.1. Bash shell:

http://www.bashoneliners.com

https://ss64.com/bash

9.2.2. GNU / Linux Help System:

https://www.kernel.org/doc/man-pages

https://www.gnu.org/manual

9.2.3. UNIX operating system resources:

https://www.grymoire.com/Unix

9.2.4. Practices for using commands in GNU / Linux:

http://www.commandlinefu.com

Syllabus of the course

Is designed by teacher Associate Professor, Ph.D, Oleksandr Rokovyi

Adopted by Department of Computing Engineering (protocol № 13, 10 May 2023)

Approved by the Faculty Board of Methodology (protocol № 11, 29 June 2023)