**"Kyiv Vocational College of Communication"**

**Cyclic Commission of Computer Engineering**

**EXECUTION REPORT**

**LABORATORY WORK No. 2**

from the discipline: "Operating systems"

**Topic: "Getting to know the interface and capabilities of the Linux OS"**

**Performed by students of the group:**

Барабаш Матвій

Погребняк Ілля

Целуйко Станіслав

**Checked by the teacher**

Sushanova V.S

**Work of group students КСМ-13Б Team:** **PMC wolf group**

**Kyiv 2023**

**The goal of the work:**

1. Familiarity with Linux OS interfaces.

2. Obtaining practical skills of working in Linux OS environments and mobile OS - their graphical ones shell, logging in and out of the system, familiarization with the structure of the desktop, study basic actions and settings when working in the system

**Material provision of classes**

1. IBM PC type computer.

2. OS family Windows (Windows 7).

3. Virtual machine - Virtual Box (Oracle).

4. GNU/Linux operating system - CentOS.

5. Cisco network academy site netacad.com and its online Linux courses.

**Progress:**

1. Working in graphical mode in the Linux family OS (working with Internet sources):

**1.1** The "Applications" tab is one of the main components of the user's workspace in many operating systems and work environments. This tab is designed to access various programs and applications that the user can use to perform various tasks. The main components of the "Applications" tab may include:

1) "Start" or "Applications" menu:

- This is the main menu, which is usually located at the bottom or top of the screen.

- The menu contains icons or lists of programs and applications that are available to the user.

- Some operating systems may also provide the ability to search for programs in this menu.

2) Shortcuts to popular programs:

- These are usually icons of applications or programs that the user uses most often.

- They can be located directly on the desktop or in special quick access panels.

3) File manager:

- The Applications tab may also contain a file manager that allows the user to organize and work with files and folders on the computer.

4) Program categories:

- Applications can be divided into categories such as "Office applications," "Games," "Multimedia," "Internet," etc.

- User can browse applications divided into these categories for easy search.

5) Search and filtering:

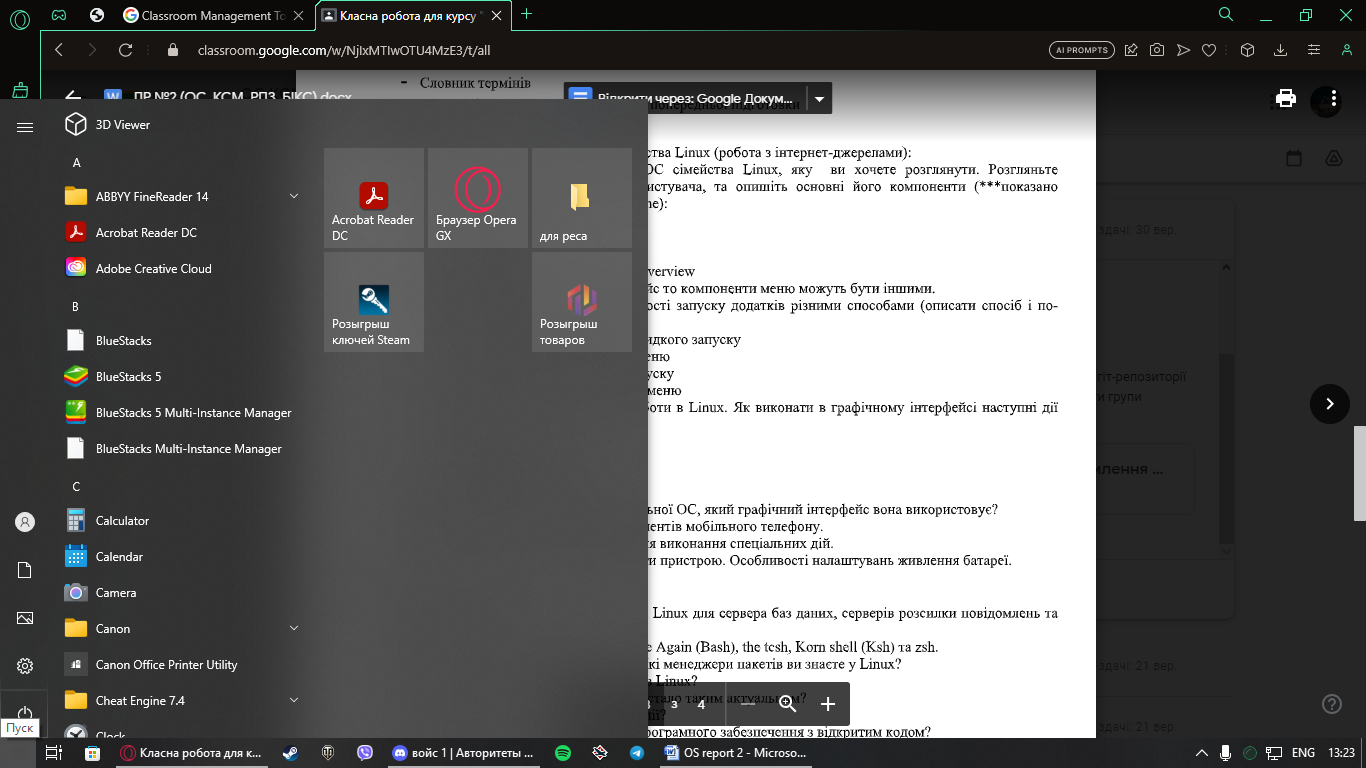
- It is usually possible to search for programs by name or keywords.

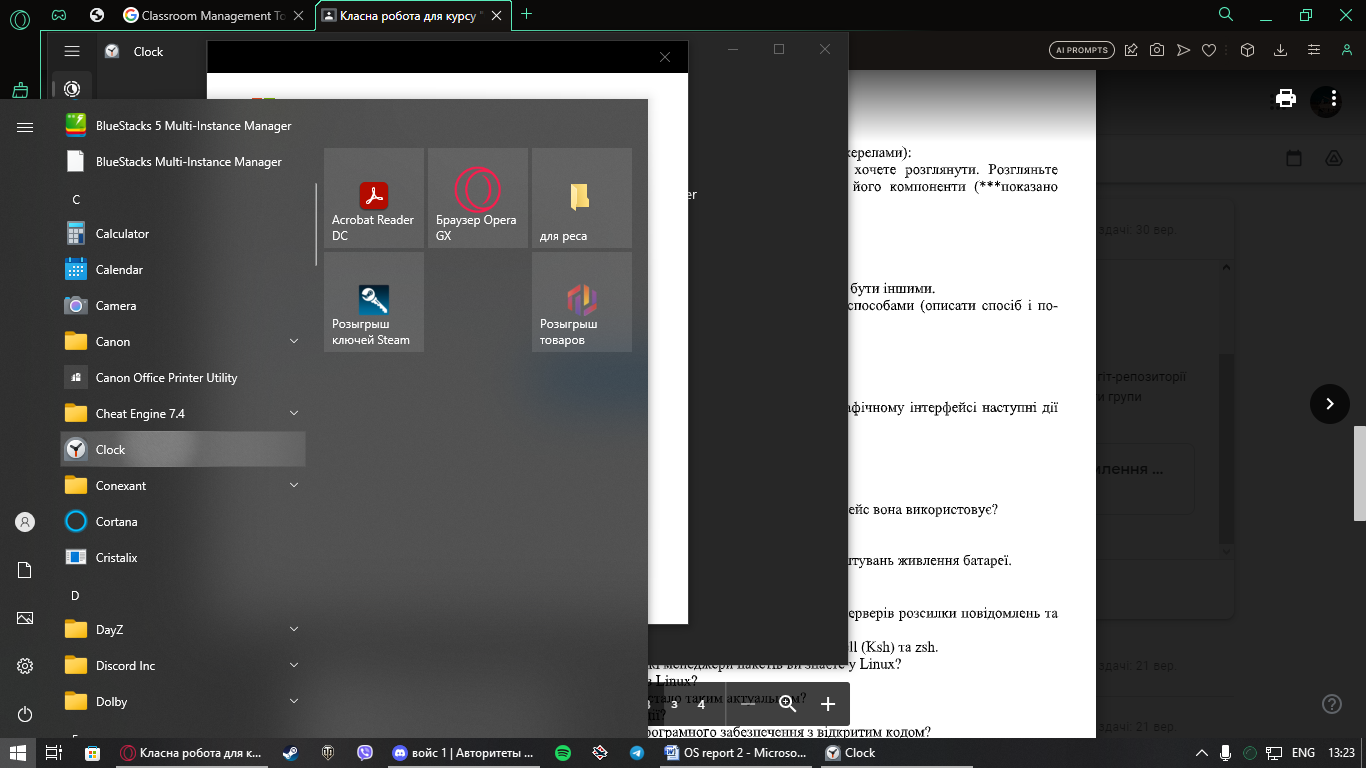
- Some operating systems provide filters to sort programs by various criteria, such as alphabetical order, update date, etc.

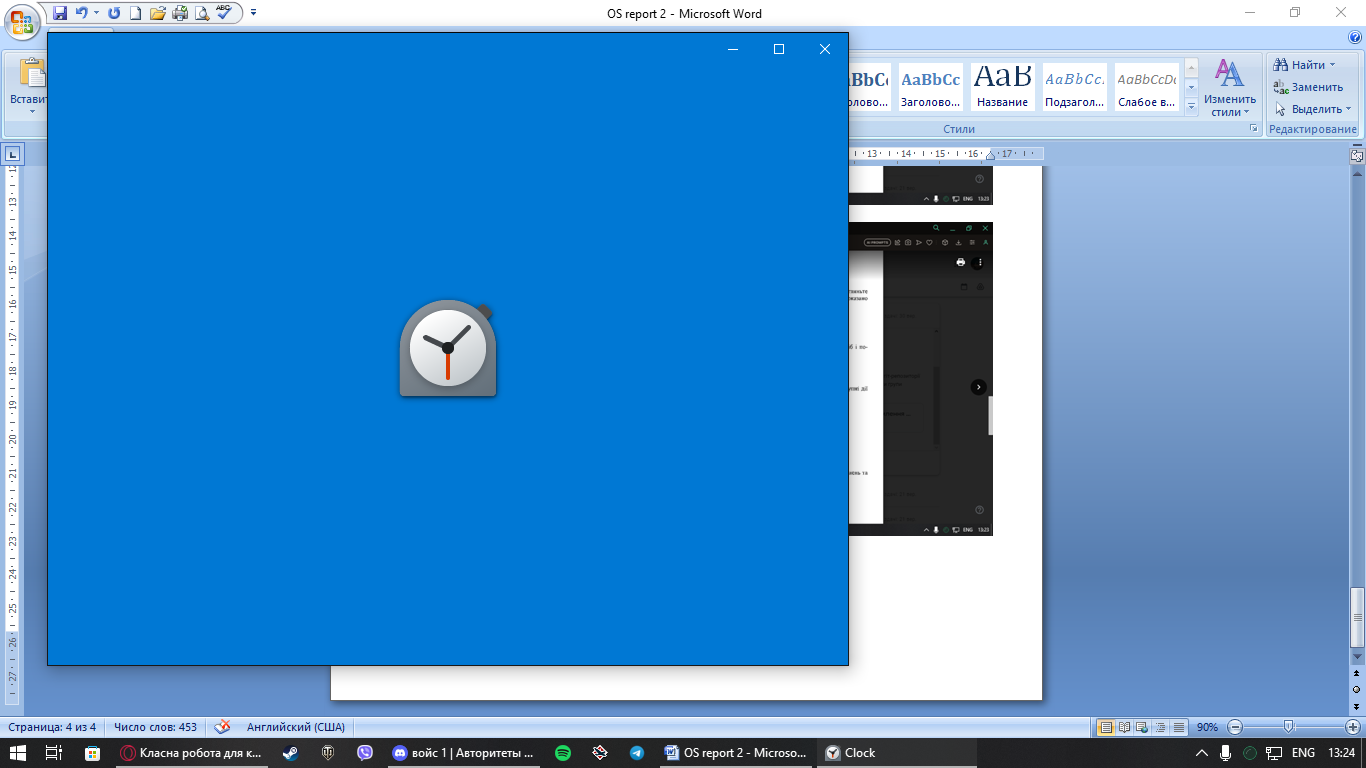
6) Additional tools:

- Depending on the operating system and settings, other tools may be available in the user workspace, such as system settings, control panel, security tools, etc.

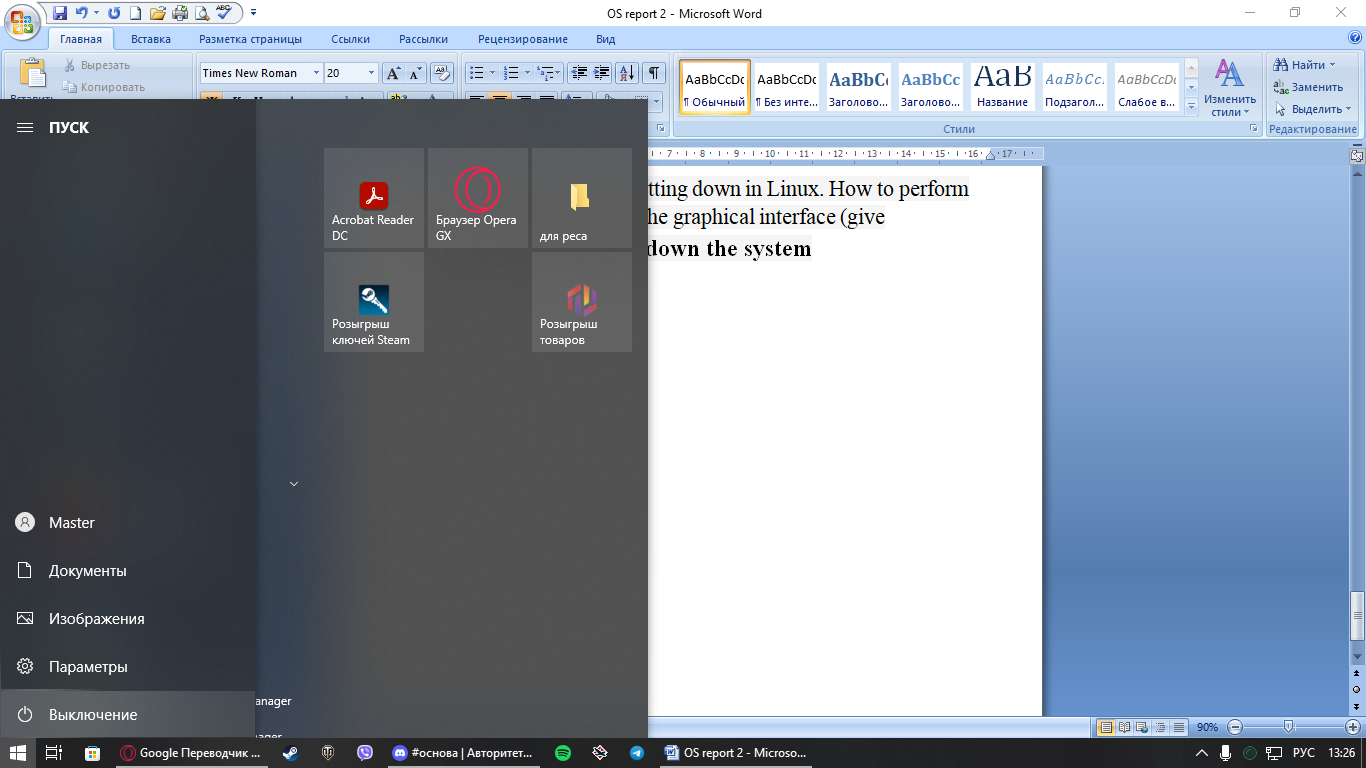
**1.2 Launching programs. Explore the possibilities of launching applications in different ways (describe the method and how ability to show screenshots):**

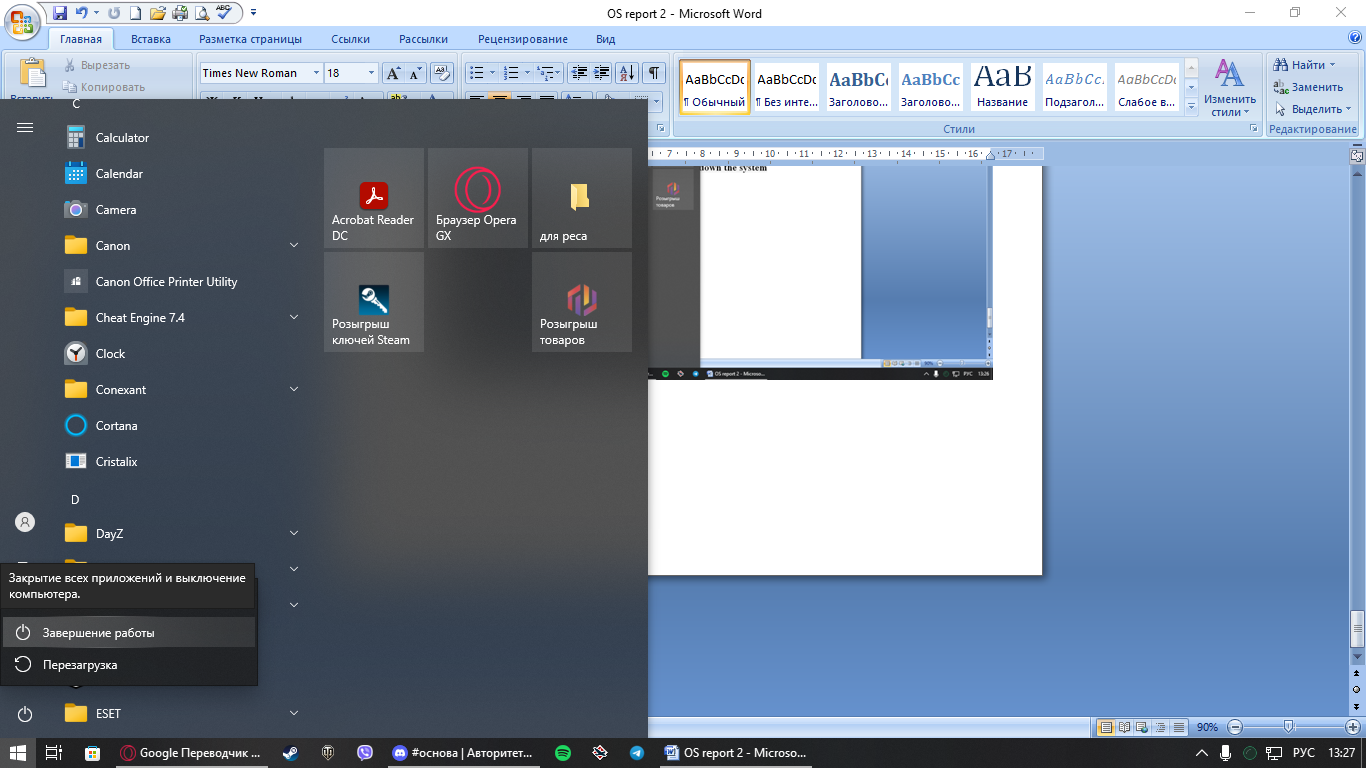
****





**1.3**. Logging out and shutting down in Linux. How to perform the following actions in the graphical interface (give screenshots): **Shutting down the system**





**2. Work in a mobile OS environment**.

**2.1)** Android 10 is an operating system for mobile devices developed by Google. The main interface of Android 10 includes several key elements that distinguish the main menu and make it convenient for users. Here is a brief description of the main menu and its components:

- Home screen: The Android 10 home screen is the starting point that appears after unlocking the device. You can customize it by adding widgets, shortcuts, and shortcuts to the apps and features you need.

- Notifications and quick settings panel: By swiping down from the top of the screen, you open the notifications and quick settings panel. This panel displays message notifications, device status, and allows you to quickly enable or disable modes such as Wi-Fi, Bluetooth, Do Not Disturb, and more.

- Apps bar: Accessing the apps bar is done by tapping the "See all apps" icon on the home screen, or usually by swiping up from the app icon. The apps bar displays all installed apps in alphabetical order, and you can search and run them from here.

- "Back" and "Recent" button: System navigation elements are usually located around these buttons. The "Back" button is designed to go back in an application or menu, and the "Recent" button allows you to view recently used applications and switch between them.

- Google Search: The home screen usually has a Google search widget that allows users to search the Internet or other sources directly from the home screen.

- Settings subsystem: Access to the main settings of the device is usually done through the Settings menu, which can be found in the application tray or quick settings. This menu allows users to configure various device settings, including network connections, sound, screen, and security.

**2.2)** The settings menu on a mobile phone is a place where the user can configure various parameters and functions of the device. This menu usually contains different sections, where each section is responsible for a certain category of settings. Here's a general description of the sections and components you'll typically find in a mobile phone's settings menu:

- Wi-Fi and network: In this section, you can configure the connection to the Wi-Fi network, mobile Internet, Bluetooth and other network parameters. Here you can also view data usage information and configure roaming.

- Sound and vibration: In this section, you can change the volume, choose different sound profiles, set vibration, change sounds for notifications and calls.

- Display and brightness: Here you can adjust screen parameters such as brightness, automatic adaptation to light, night view mode and screen display settings.

- Requests and accounts: In this section you can add or manage Google accounts, e-mail, social networks and other accounts. Here you can set restrictions for children and family settings.

- Applications and notifications: Here you can manage applications, install them, allow or deny notifications from individual applications, as well as configure permissions for applications.

- System settings: General system settings such as date and time, language and input, data backup and reset, software updates and other options are usually located in this section.

- About phone: This section contains information about the device, including battery status, memory size, operating system version and other technical information.

- Security and privacy: Here you can set device security such as fingerprint, password or PIN, and manage privacy settings such as app permissions and private data security.

- Storage and USB: In this section you can manage data storage, manage memory cards and connected USB devices.

- Accounts: This section allows you to add and configure user accounts, including Google and others.

**2.3.** Using keyboard shortcuts to perform special actions is a fairly common practice on computers and mobile devices. Some of the most common keyboard shortcuts for various operations include the following:

- Ctrl + C (or Command + C on Mac): Copy the selected text or object to the clipboard.

- Ctrl + X (or Command + X on Mac): Cut the selected text or object to the clipboard.

- Ctrl + V (or Command + V on Mac): Paste content from the clipboard to the current location.

- Ctrl + Z (or Command + Z on Mac): Undo the last action. - Ctrl + Y (or Command + Y on Mac): Redo the undone action.

- Ctrl + S (or Command + S on Mac): Save the current document or file.

- Ctrl + P (or Command + P on Mac): Print the current document or page.

- Alt + Tab (or Command + Tab on Mac): Switch between windows or programs.

- Ctrl + Alt + Delete (or Command + Option + Escape on Mac): Call the task management menu or task manager to close programs or restart the computer.

- Ctrl + F (or Command + F on Mac): Search for text on a page or document.

- Ctrl + N (or Command + N on Mac): Open a new window or a new document. - Ctrl + W (or Command + W on Mac): Close the current window or tab.

- Ctrl + T (or Command + T on Mac): Open a new tab in the web browser.

- Ctrl + Shift + Esc (or Command + Option + Esc on Mac): Open Task Manager without a menu in Windows.

- Ctrl + Shift + T (or Command + Shift + T on Mac): Restore the last closed tab in the web browser.

**2.4.** **Logging in and shutting down your device, as well as battery power settings, vary by operating system and device, but the basic concepts and actions remain the same. Below is a general description of these operations:**

- Login to the system: PIN or password: When turning on the device or after a period of inactivity, the user usually needs to enter their PIN or password to log in.

Some devices also support fingerprint recognition or facial recognition for authentication.

Fingerprint or facial recognition (Face ID, Face Unlock): Some devices support biometric authentication.

User can set fingerprint or facial recognition for fast and secure login.

Guest Mode: Some devices allow you to create guest accounts or guest mode, which restricts access to basic features and data on the device.

- Shutting down the device:

Closing applications: To end work with applications, the user can simply close them by pressing the Close button or using the close gesture.

Shutting down the device: To turn off the device completely, the user can hold down the Power button and select "Shutdown" or "Reboot".

Sleep Mode or Screen Lock: During inactivity, the device can go into sleep mode or screen lock to save power and protect data.

- For effective battery power management on mobile devices, you can use various settings that will allow you to extend the duration of the device's operation on a single charge. Here are some basic battery power settings:

- Power saving mode: Many mobile devices have power saving modes that limit the functionality of the device to save battery power. These modes can turn off background processes, sync, vibrate, and more.

- Automatic brightness: Enabling automatic adjustment of the screen brightness allows the device to adapt the brightness depending on the ambient lighting.

- Data sync: Turning off automatic sync for accounts and apps can reduce data usage and save battery life.

- Hardware: You can turn off features like Bluetooth, NFC, GPS when not in use, and turn off vibration if needed.

- Schedules and timings for turning on and off: Some devices have the ability to set schedules for turning off and turning on the device at a certain time, which allows you to save charge at night or during a break in work.

- Notifications and background apps: Manage app notification permissions and prevent power-hungry apps from running in the background. - Software Updates: Always keep your device up-to-date as new versions of the operating system may include optimizations to save battery life.

- Battery health: Some devices provide information about battery health and battery usage in the settings.

- Chargers: Use original or recommended chargers and cables, as improper charging may affect battery life.

**Control questions**

**1)** Given the variety of server applications for Linux, here are examples of some popular applications for database servers, messaging servers, and file servers:

**- Database servers:**

\* MySQL / MariaDB: MySQL and its fork MariaDB are very popular open source database servers. They support SQL and are used to store and manage data.

\* PostgreSQL: PostgreSQL is a powerful relational database management system (RDBMS) with high reliability and scalability. It supports SQL extensions and has a large user community.

\* MongoDB: MongoDB is a NoSQL database that uses documents in JSON format. It allows you to store structured and unstructured data, and is particularly suitable for projects with a large number of documents.

**- Messaging servers:**

\* Postfix: Postfix is an SMTP server used to send and receive emails. It is easy to configure and provides a high level of security.

\* Exim: Exim is another popular SMTP server that provides the ability to handle email on the server.

\* Sendmail: Sendmail is one of the oldest MTAs (Mail Transfer Agent) that can be used to handle mail traffic.

**- File sharers:**

\* Samba: Samba allows Linux servers and Windows computers to share files and print products. It is the most popular solution for integrating Linux and Windows on the same network.

\* NFS (Network File System): NFS is a protocol and file system for distributed access to files in Unix-like operating systems. It allows you to share files between different Linux servers and computers.

\* OwnCloud / Nextcloud: OwnCloud and Nextcloud are solutions for creating your own cloud storage and file sharing. They provide the ability to store files, synchronize them between different devices and securely manage access to them.

**2)** Shells in Unix and Linux systems are command shells that allow users to interact with the operating system using text commands. Here is a comparison of some of the most popular shells: Bourne, C, Bourne Again (Bash), tcsh, Korn shell (Ksh), and zsh.

- Bourne Shell (sh):

\* History: The first shell created by Stephen Bourne in 1979. - Features:

\* Simple and minimalistic.

\* Few built-in functions for automation.

\* Usage: Usually used as a system shell, sometimes as a command provider for scripts. - C Shell (csh):

\* History: Created by Bill Joy in 1978.

- Features: \* Has a similar syntax to the C programming language.

\* Supports command history and shell command mode.

- Usage: Little used in modern systems due to limited capabilities. - Bourne Again Shell (Bash):

\* History: An enhanced version of the Bourne Shell, created as a standard shell for many systems.

- Features: \* Full compatibility with Bourne Shell.

\* Greatly advanced capabilities such as shell control, signal processing, environment variables and more.

\* Usage: The most popular shell in modern Unix and Linux systems.

- tcsh:

\* History: An enhanced version of the C Shell, created as a reaction to the limitations of the C Shell.

- Features:

\* Improved command history.

\* Full compatibility with C Shell.

- Usage: Used as a shell with much more advanced features for users who prefer the C Shell syntax.

- Korn Shell (Ksh):

\* History: Created by David Korn in 1983.

- Features:

\* Full compatibility with Bourne Shell.

\* Built-in functions for scripts and working with files and strings.

\* Usage: Used as an alternative to Bash and on systems that support it.

- Zsh (Z Shell):

\* History: Created by Paul Falstad in 1990.

- Features:

\* A large number of extensions and plugins.

\* Advanced capabilities of autocomplete and interactive work.

\* Usage: Used by active users looking for a multi-functional shell with significant customization options.

**3)** A package manager is a tool in operating systems that allows users to easily install, update, remove, and manage software packages (programs) on their computer. The main purpose of a package manager is to simplify the process of installing and managing programs, to ensure consistency and control over program versions, and to allow automatic resolution of dependencies between packages. Here are some popular Linux package managers:

\* APT (Advanced Package Tool):

- Used in Debian and Ubuntu distributions.

- Commands: apt-get, apt, apt-cache.

- Packages in .deb format.

\* YUM (Yellowdog Updater Modified):

- Used in Red Hat, CentOS and Fedora distributions.

- Commands: yum. - Packages in .rpm format.

\* DNF (Dandified YUM):

- Used in Fedora from version 22 and other distributions that switched to DNF.

- Teams: dnf.

- Packages in .rpm format. Pac Man:

\* Used by Arch Linux and other distributions that use an Arch-like package management system.

- Commands: pacman.

- Packages in .pkg.tar.xz format.

\* Zypper:

- Used in openSUSE and SUSE Linux Enterprise distribution. - Commands: zypper.

- Packages in .rpm format.

- Portage (Gentoo):

\* Used in the Gentoo distribution.

- Commands: emerge.

- Does not have packages in .deb or .rpm format; installs programs from source code.

\* Snap and Flatpak: These are two conceptually distinct package managers that allow you to install and use applications with isolated environments and standalone dependencies. Supported on various distributions. Condo: It is used to manage packages of Python and other programming languages. Popular in scientific and analytical fields.

**4)** - Linux has a rich set of tools and security tools that help protect the system from various threats and ensure a high level of security. Here are some of the basic security tools in Linux:

- Access rights (File Permissions): Linux uses a system of access rights to files and directories, which determines which users and groups can read, write and execute files. This is an important means of limiting access to sensitive information.

- Software firewall (Firewall): In Linux, you can configure a software firewall, such as iptables or the more modern firewalld, to monitor network traffic and filter packets. This helps protect the system from unauthorized access over the network.

- SELinux and AppArmor: Mandatory access control systems (MAC) such as SELinux (Security-Enhanced Linux) and AppArmor allow detailed configuration of rules for access to resources on the system, which increases security.

- System updates: Regular software and operating system updates are important to eliminate vulnerabilities and improve system security. Linux distributions provide mechanisms for updating programs and packages.

- Antivirus software: Although Linux is less vulnerable to viruses compared to other operating systems, there are antivirus programs to detect and clean viruses and malware.

- Logging system: Logging of events and actions on the system helps to detect unauthorized access and vulnerabilities. Business Linux systems often use syslog and other logging tools.

- Security Auditing: Security auditing tools such as OpenSCAP allow you to check the system for compliance with security standards and identify potential problems.

- Encryption: Using encryption to protect data at the level of files, disk space, and network traffic helps prevent unauthorized users or attackers from accessing data.

- Authentication systems: Linux supports various authentication systems, such as PAM (Pluggable Authentication Module), which allow you to set authentication rules for users.

- Execution Permissions: Setting execution permissions for files and directories helps to avoid running dangerous programs.

- Access audit: This mechanism allows you to keep track of user access to system resources and objects.

**5)** The use of virtualization has become relevant for a number of reasons that contribute to the popularity of this technology: Efficient use of resources, Isolation and security, Convenient testing and development, Easy migration and backup, Reduction of equipment costs, Easy scaling, Work in cloud services, Support for various operating systems , Dynamic allocation of resources.

**6)** Containerization is a methodology for deploying and managing software applications and their dependencies in a virtual environment known as a container. Containers are defined as independent, isolated, and easy-to-use packages that include all the necessary components to run an application, including application code, libraries, runtime, and configuration.

**7)** Using open source software has its advantages and disadvantages that you can consider when choosing this type of software product. Here are some of them:

- Advantages of using open source software:

Cost: A big advantage is the absence of costs for purchasing licenses. You are free to use, distribute and modify open source code. Access to the source code: You have the ability to view and modify the source code of the application, which allows you to customize the software to your needs and fix errors.

Developer Community: Open source software typically has a large and active community of developers and users who provide support, solve problems, and develop the product.

High security: Many open-source projects are heavily vetted for security because the world has access to the source code and can identify potential vulnerabilities. Supplier Independence: You are not dependent on a specific manufacturer or supplier. You have control over the software.

Modularity and extensibility: Open source software usually has a modular architecture that allows extensions and plugins to be added.

License freedom: Most open licenses allow you to use the software without restrictions on the number of copies, distribution, and commercial use.

- Disadvantages of using open source software:

Lack of guaranteed support: In some cases, professional support and quality assurance may be limited or absent.

Difficulty in choosing: With so many alternatives, it can be difficult to choose the right project that meets your needs.

Possibility of multi-severability: Since applications are developed by a community, if the project loses support, it can lead to security and compatibility issues.

Lockedness in open formats: Some open applications may use file formats or protocols that are not always compatible with other applications. Resource Requirement: Some open applications may be demanding on computing hardware resources.

**Conclusion**

Important practical skills and knowledge were gained while studying and working in Linux and mobile OS environments. The main conclusions are as follows: Work in Linux OS: The GUI of a Linux OS (such as Ubuntu) and its main elements such as the taskbar, menus and desktop were introduced. Learned how to log in using username and password and exit. The file system structure and basic actions with files and directories were considered. Work in mobile OS: The graphical user interface of the mobile OS (e.g. Android 10) and its elements such as the home screen, the quick access bar and the settings menu were investigated. Learned how to log into a mobile device using a PIN, unlock the device, and exit. The basic settings of the device, such as Wi-Fi, sound, application and security settings, were familiarized. This work provided important skills in the chosen environment and increased understanding of Linux operating systems and mobile OSes. Overriding is the importance of knowledge and skills in the field of working with computers and mobile devices in today's world, where these OS are widespread and important for work and personal use.