“Київський фаховий коледж зв’язку”

Циклова комісія комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №2**

**з дисципліни: «Операційні системи»**

Тема: “Знайомство з інтерфейсом та можливостями ОС Linux”

Виконавли студенти

групи КСМ-23б

Команда VelikieBomji:

Бережний Т.І., Лобода Д.Р.,

Чухманенко М.В

Перевірила викладач

Сушанова В.С.

Київ 2024

**Завдання для попередньої підготовки.**

1. \*Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.
2. Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

* Chapter 3 - Working in Linux
* Chapter 4 - Open Source Software and Licensing

1. Пройдіть тестування у курсі NDG Linux Essentials за такими темами:

* Chapter 03 Exam
* Chapter 04 Exam

1. \*\*Дайте визначення наступним поняттям:

* CLI-режим

Це режим командної строки

* Термінал на основі графічного інтерфейсу користувача

A GUI terminal is a program within the GUI environment that emulates a terminal window.

* Віртуальний термінал

A virtual terminal can be run at the same time as a GUI but requires the user to log in via the virtual terminal before they can execute commands (as they would before accessing the GUI interface).

1. Підготувати в електронному вигляді початковий варіант звіту:

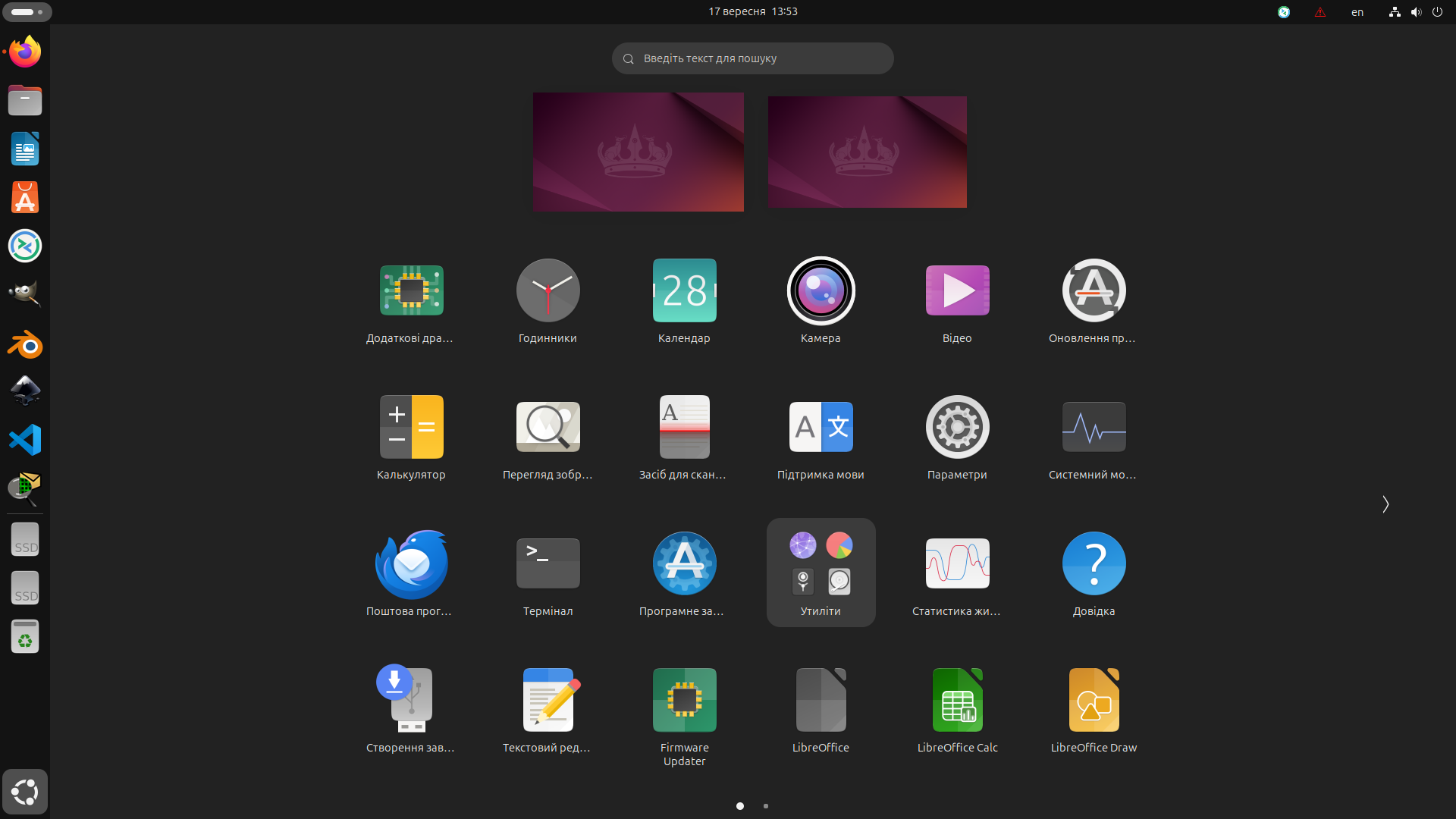
* Титульний аркуш, тема та мета роботи
* Словник термінів

|  |  |
| --- | --- |
| Термін українською | Термін англійською |
| GUI terminal | Віртуал графічного інтерфейсу |
| Virtual terminal | Віртуальний термінал |
| Application | Застосування |
| CPU | Центральний процесор |
| GPU | Графічний процесор |
| Server applications | Серверні програми |
| Desktop applications | Настільні програми |
| Tools | Інструменти |
| Open source | Відкритий код |

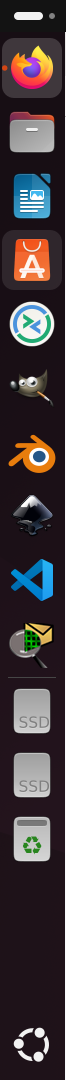
* Відповіді на п.4 з завдань для попередньої підготовки

**Хід роботи.**

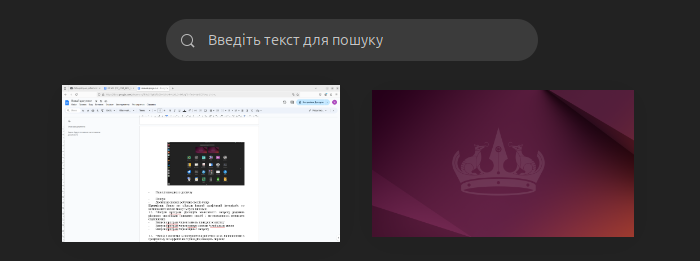
* 1. Робота в графічному режимі в ОС сімейства Linux (робота з інтернет-джерелами):
  2. Оберіть графічну оболонку для ОС сімейства Linux, яку ви хочете розглянути ***(в 401 ауд. це Gnome)***. Розгляньте структуру робочого простору користувача, та опишіть основні його компоненти:
* Основне меню



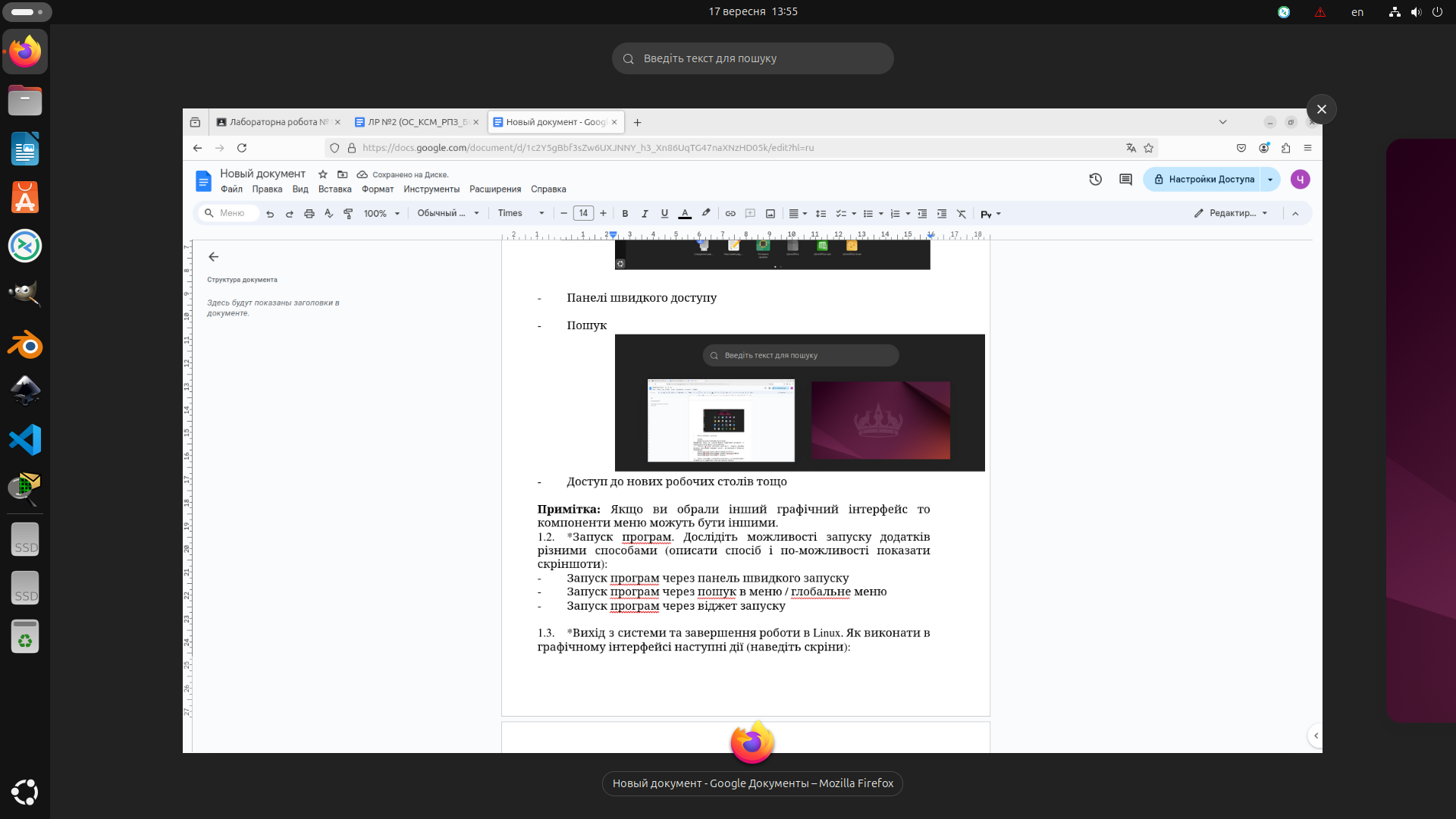
* Пнелаі швидкого доступу



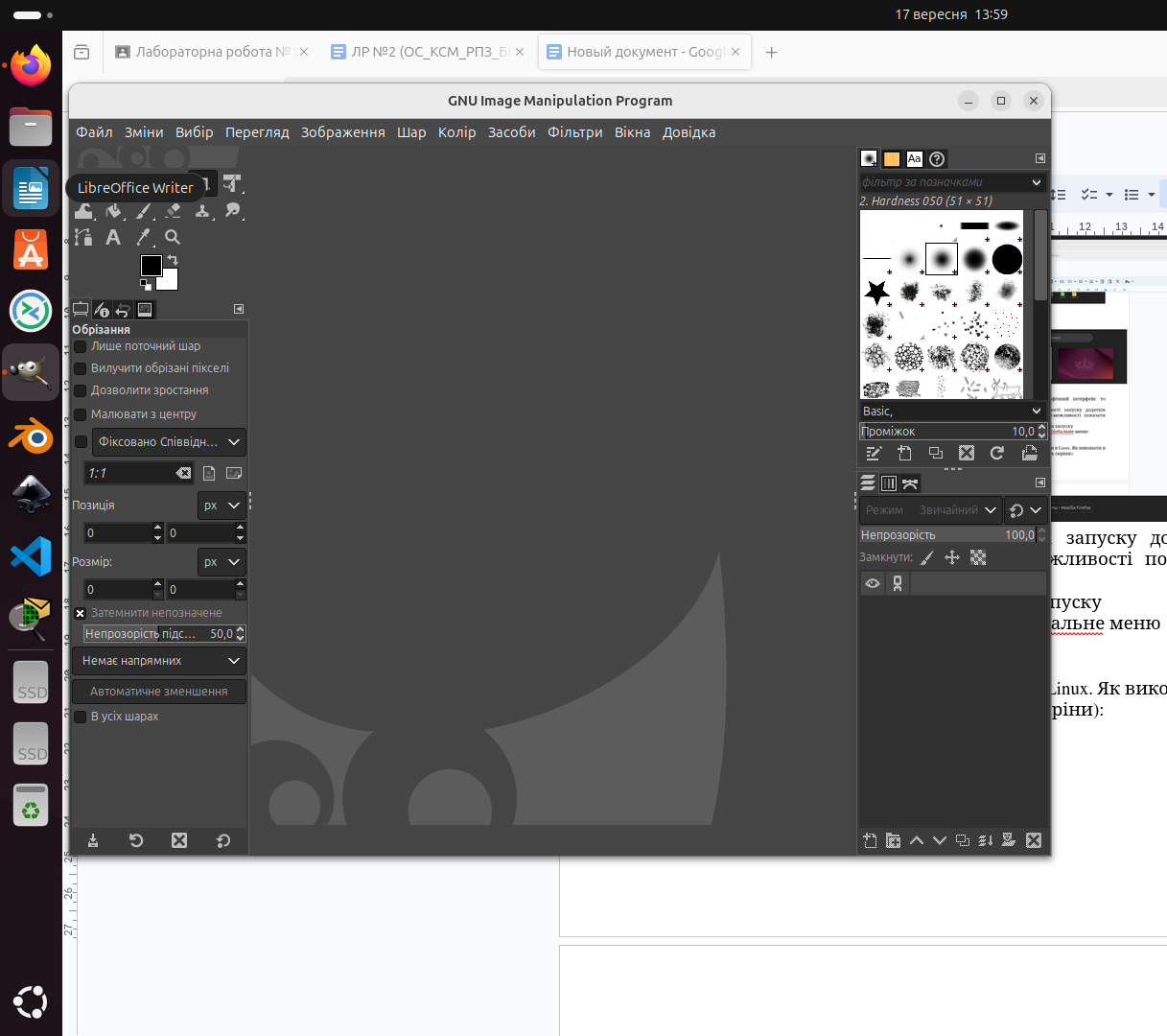
* Пошук



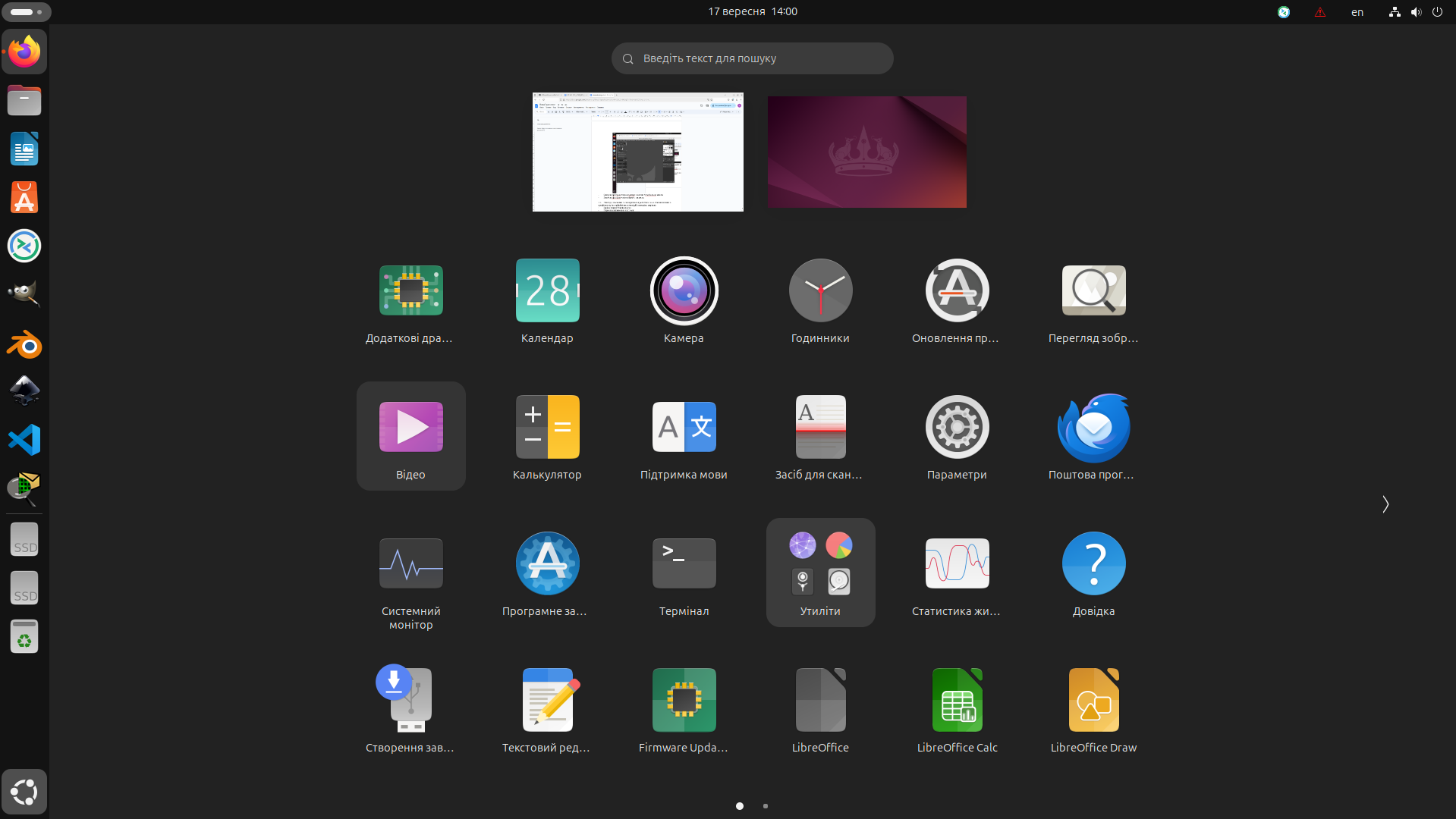
* Доступ до нових робочих столів тощо



* 1. \*Запуск програм. Дослідіть можливості запуску додатків різними способами (описати спосіб і по-можливості показати скріншоти):
* Запуск програм через панель швидкого запуску



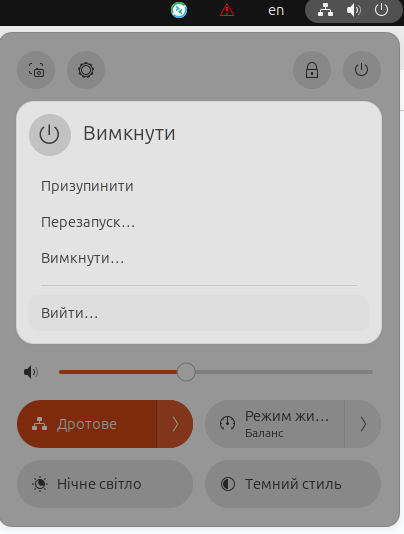
* Запуск програм через пошук в меню / глобальне меню



* Запуск програм через віджет запуску



* 1. \*Вихід з системи та завершення роботи в Linux. Як виконати в графічному інтерфейсі наступні дії (наведіть скріни):
* Зміна користувача на root
* Перезавантаження системи
* Вимкнення системи



1. \*\*Робота в середовищі мобільної ОС. (Бережний Т.І.)
   1. Опишіть головне меню вашої мобільної ОС, який графічний інтерфейс вона використовує?

Android (на основі модифікованого ядра Linux) — це мобільна операційна система, розроблена компанією Google. Хоча операційна система Android є безкоштовним програмним забезпеченням з відкритим кодом, більшість програмного забезпечення, що поставляється разом із ним (включаючи програми Google та програмне забезпечення, встановлене постачальником), є власним програмним забезпеченням.

Випуски Android до 2.0 (1.0, 1.5, 1.6) використовувалися виключно на мобільних телефонах. Випуски Android 2.x в основному використовувалися для мобільних телефонів, але також і для деяких планшетів, Android 3.0 був орієнтованим на планшети і офіційно не працює на мобільних телефонах, тоді як сумісність телефону та планшета була об'єднана з Android 4.0. Поточна версія Android — Android 15, випущена 3 вересня 2024 року.

Android використовує графічний інтерфейс, що називається Material Design. Головне меню (якщо мова йде про домашній екран) зазвичай складається з ряду додатків і віджетів, розташованих на різних панелях. Користувач може перегортати ці панелі горизонтально або вертикально, залежно від налаштувань. На головному екрані також можуть бути розміщені швидкі доступи до популярних додатків, пошукова строка або інші інструменти, які користувач встановлює чи налаштовує за своїми уподобаннями.

* 1. Опишіть меню налаштувань компонентів мобільного телефону.

Мережі та підключення:

* Wi-Fi – увімкнення/вимкнення Wi-Fi, пошук і підключення до доступних мереж.
* Мобільна мережа – налаштування мобільних даних, вибір оператора, тип мережі (2G, 3G, 4G, 5G).
* Bluetooth – увімкнення/вимкнення, пошук та підключення до пристроїв.
* VPN – налаштування та підключення віртуальної приватної мережі.
* Точка доступу і модем – створення точки доступу Wi-Fi для роздачі інтернету з мобільного пристрою.

Пристрої та дисплей:

* Екран – налаштування яскравості, режиму темної теми, зміни роздільної здатності та шпалер.
* Звук – регулювання рівня гучності дзвінка, медіа, сповіщень, а також налаштування вібрації і тонів виклику.
* Сенсорний екран – калібрування сенсорного екрана, налаштування жестів і сенсорної чутливості.
* Повідомлення та активний рядок – налаштування повідомлень.
* Робочий стіл – налаштування робочого столу.

Батарея:

* Режими енергозбереження – керування режимами для збільшення часу роботи акумулятора.
* Використання батареї – перегляд додатків, які найбільше споживають заряд акумулятора.

Безпека та конфіденційність:

* Блокування екрану – вибір методу блокування: пін-код, графічний ключ, відбиток пальця, обличчя.
* Доступ до даних – налаштування дозволів для додатків.
* Зашифрування телефону – увімкнення шифрування даних на пристрої.
* Відбиток, данні лиця та захист пристрою – розблокувати телефон за допомогою сканеру пальця, лиця або паролю.
* Безпека та екстрені служби – повідомлення про небезпеку та при можливості швидко викликати на допомогу

Система:

* Оновлення системи – перевірка та встановлення оновлень для операційної системи.
* Мова і введення – вибір мови інтерфейсу, налаштування клавіатури, голосового введення.
* Резервне копіювання і скидання – налаштування автоматичного резервного копіювання та скидання пристрою до заводських налаштувань.
  1. Використання комбінацій клавіш для виконання спеціальних дій.
* Скріншот **- кнопка живлення + кнопка зменшення гучності.**
* **Перезавантаження пристрою - кнопка живлення** (натиснути й утримувати кілька секунд).
* Перехід у режим завантаження (bootloader) **- кнопка живлення** + **кнопка зменшення гучності** (утримувати при вимкненому пристрої).
* Активування Google Assistant **- утримання бокової кнопки живлення.**
* Відновлення системи (Recovery Mode) **- кнопка живлення** + **кнопка збільшення гучності** (на вимкненому пристрої).
  1. Вхід у систему та завершення роботи пристрою. Особливості налаштувань живлення батареї.

Для того, щоб ввійти в систему потрібно:

* Ввімкнути пристрій (кнопка живлення).
* Розблокувати екран (свайп вверх), якщо є пароль ввести його, розблокувати через сканер пальця або лиця.

Для того, щоб вийти з системи достатньо натиснути на кнопку живлення і пристрій вимкнеться.

Особливості налаштувань батареї на Android включають режими енергозбереження, які обмежують активність додатків у фоновому режимі для зменшення споживання енергії. Також є адаптивна батарея, яка вивчає звички користувача і оптимізує роботу додатків для продовження часу роботи пристрою.

**Контрольні запитання (Лобода Д.Р.)**

1. Наведіть приклади серверних додатків Linux для сервера баз даних, серверів розсилки повідомлень та файлообмінників.

MySQL/MariaDB: Популярні реляційні бази даних з відкритим кодом. MySQL широко використовується в різних проєктах, а MariaDB є її форком із більш відкритою ліцензією.

PostgreSQL: Потужна об'єктно-реляційна система керування базами даних, відома своєю надійністю та розширеними можливостями.

SQLite: Легка база даних, що не потребує окремого серверного програмного забезпечення і добре підходить для вбудованих рішень або невеликих проєктів.

1. Порівняйте оболонки Bourne, C, Bourne Again (Bash), the tcsh, Korn shell (Ksh) та zsh.

1.Bourne shell (sh):

переваги: Основна оболонка Unix до появи інших, Підтримує прості функції скриптингу та управління процесами.

Недоліки: Обмежений функціонал у порівнянні з сучасними оболонкамию, Не підтримує розширені функції, такі як історія команд або автодоповнення.

2. C shell (csh):

переваги: Подібний до C-синтаксис для умовних конструкцій і циклів, Підтримує історію команд, редагування команд і аліаси (короткі команди).

Недоліки: Не така ефективна для скриптів, як Bourne shell, і має деякі баги в обробці команд.

3. Bourne Again Shell (Bash):

Переваги: Найбільш потужна і зручна для більшості користувачів Linux.

Недоліки Bash: Повільніший порівняно з іншими оболонками, Мінімальні можливості для налаштування порівняно з zsh, Не підтримує глобальні аліаси.

4. tcsh:

Переваги: підтримує автодоповнення, редагування командної строки та більш потужну обробку історії, більш дружелюбний до користувача інтерфейс у порівнянні з C shell.

Недоліки: Як і C shell, не найкращий вибір для скриптингу.

5. Korn shell (ksh):

Переваги: Висока продуктивність і потужні можливості для програмування.

Недоліки: Менш популярна в сучасних Linux-системах через конкуренцію з Bash.

6. Z shell (zsh):

Переваги: надзвичайно потужна та гнучка, з великою кількістю опцій для налаштування, підтримує автодоповнення, історію команд, глобальні аліаси, підсвічування синтаксису, вбудовані можливості для сценаріїв, інтерактивне середовище дуже настроюване,надзвичайно багатофункціональна і гнучка, може бути налаштована під будь-які потреби.

Недоліки: Може бути складною для новачків через багатий набір функцій і необхідність налаштування.

1. Для чого потрібен менеджер пакетів. Які менеджери пакетів ви знаєте у Linux?

Менеджер пакетів — це програма, яка автоматизує процеси встановлення, оновлення, налаштування та видалення програмного забезпечення в операційній системі. У Linux менеджери пакетів відіграють ключову роль у підтримці системи в актуальному стані та забезпеченні встановлення необхідного ПЗ разом із залежностями. Основні менеджери пакетів у Linux

Існує кілька популярних менеджерів пакетів, які використовуються в різних Linux-дистрибутивах. Ось кілька основних:

1. APT (Advanced Package Tool)

Використовується у: Debian, Ubuntu та їх похідних дистрибутивах.

Команди:

apt install package: Встановлення пакета.

apt update: Оновлення списку пакетів.

apt upgrade: Оновлення вже встановлених пакетів.

apt remove package: Видалення пакета.

Особливості: Працює з форматами .deb, підтримує великі репозиторії програмного забезпечення та автоматично керує залежностями.

2. DNF (Dandified YUM)

Використовується у: Fedora, RHEL, CentOS.

Команди:

dnf install package: Встановлення пакета.

dnf update: Оновлення системи.

dnf remove package: Видалення пакета.

Особливості: Наступник YUM, працює з RPM-пакетами і пропонує кращу продуктивність та вирішення залежностей.

3. YUM (Yellowdog Updater, Modified)

Використовується у: Старіші версії Fedora, CentOS, RHEL.

Команди:

yum install package: Встановлення пакета.

yum update: Оновлення системи.

yum remove package: Видалення пакета.

Особливості: Старіший менеджер для роботи з RPM-пакетами, зараз переважно замінений DNF.

4. RPM (Red Hat Package Manager)

Використовується у: Fedora, CentOS, RHEL, openSUSE.

Команди:

rpm -i package.rpm: Встановлення пакета.

rpm -e package: Видалення пакета.

Особливості: Використовується для керування пакетами у форматі .rpm, але не вирішує залежності автоматично (для цього потрібен YUM або DNF).

1. \*Які засоби безпеки використовуються в Linux?
2. \*Чому використання віртуалізації зараз стало таким актуальним? (Бережний Т.І.)

Використання віртуалізації стало актуальним через її численні переваги в ефективності та економії ресурсів.

По-перше, вона дозволяє запускати кілька віртуальних машин на одному фізичному сервері, що значно знижує витрати на обладнання.

По-друге, віртуалізація спрощує управління ІТ-інфраструктурою, полегшуючи створення, налаштування і міграцію систем, а також підвищує безпеку та ізоляцію середовищ для розробки, тестування й продакшн-завдань.

1. \*Як ви розумієте поняття контейнеризації? (Бережний Т.І)

Контейнеризація – це суть сучасної розробки програмного забезпечення, де ми використовуємо ізольовані контейнери для створення, тестування і доставки додатків.

1. \*Які переваги/недоліки використання програмного забезпечення з відкритим кодом? (Бережний Т.І.)

**Переваги:**

* Дешевше в порівнянні з комерційною продукцією

За оцінками, програмне забезпечення з відкритим кодом економить власникам бізнесу 60 мільярдів доларів щороку. Це не дивно, враховуючи, що ці програми доступні кожному. Зазвичай вони пропонуються безкоштовно. Оскільки це програмне забезпечення розроблено для сумісності з будь-яким комп’ютером, воно дозволяє користувачам подовжити термін служби їх обладнання.

* Висока надійність

Поширена помилка полягає в тому, що програмне забезпечення з відкритим кодом схильне до збою. Навпаки, воно дуже надійне, оскільки перевіряється багатьма розробниками, тестувальниками та користувачами.

* Гнучкість

Оскільки програмне забезпечення з відкритим вихідним кодом не прив’язує Вас до власного продукту, Ви не обмежені певною архітектурою ІТ. Запатентовані продукти, як правило, вимагають оновлення як апаратного, так і програмного забезпечення. Таким чином, програмне забезпечення з відкритим вихідним кодом дозволяє користувачам комбінувати і поєднувати своє програмне забезпечення для створення унікальної ІТ-інфраструктури, яка відповідає їхнім потребам.

**Недоліки:**

* Уразливість до шкідливих атак

Деякі люди, які мають доступ до вихідного коду програмного забезпечення з відкритим кодом, не мають добрих намірів. Хоча більшість людей використовує такий доступ для пошуку дефектів і покращення програмного забезпечення, інші шукають способи використання вразливостей.

* На відміну від комерційного програмного забезпечення, OSS може бути не зручним для користувачів

Не кожне програмне забезпечення з відкритим кодом є зручним для користувача. Воно створене для задоволення потреб розробників у реалізації їхніх ідей. Таким чином, вони не приділяють багато уваги інтерфейсу користувача, що ускладнює його використання для тих, хто не володіє спеціальними знаннями.

* Не вистачає широкої підтримки клієнтів

Комерційно вироблене програмне забезпечення забезпечує спокій своїм користувачам. Зрештою, вони знають, хто розробив, створив і розповсюдив програмне забезпечення. Таким чином, вони знають, хто несе відповідальність, якщо програмне забезпечення не працює або спричиняє пошкодження обладнання.

**Висновок**

OSS може служити як особистим, так і діловим потребам. Це може заощадити час і гроші, навіть якщо у нього є свої недоліки, як-от відсутність спеціальної підтримки. Як і все інше, OSS має свої плюси і мінуси.

**(Лобода Д.Р.)**

1. \*\*Скільки активних віртуальних консолей (терміналів) може бути у процесі роботи Linux по замовчуванню. Як їх викликати та між ними перемикатися? Наведіть приклади?

У Linux за замовчуванням зазвичай доступно 6 активних віртуальних консолей (терміналів) для роботи в текстовому режимі і 1 консоль для графічного середовища (часто на TTY7). Це число може змінюватися в залежності від конфігурації системи, але за стандартом у більшості дистрибутивів підтримується саме 6 текстових консолей.

Як викликати віртуальні консолі та перемикатися між ними:

Перемикання між віртуальними консолями здійснюється за допомогою комбінацій клавіш:

Ctrl + Alt + Fn (де n — це номер консолі). Наприклад:

Ctrl + Alt + F1 — перемикання на першу віртуальну консоль (TTY1).

Ctrl + Alt + F2 — перемикання на другу віртуальну консоль (TTY2) і так далі до F6.

Кожна з цих консолей працює в текстовому режимі, і ви можете увійти в систему під тим же чи іншим користувачем на кожній консолі.

Приклади перемикання між консолями:

Ctrl + Alt + F1: перемикання на першу консоль (TTY1), яка відображає командний рядок. Тут ви можете увійти в систему і запускати команди.

Ctrl + Alt + F7: перемикання на графічне середовище (якщо воно активне, наприклад, GNOME, KDE).

1. \*\*Яка віртуальна консоль (термінал) виконує функцію графічної оболонки?

У більшості Linux-дистрибутивів віртуальна консоль (термінал), яка виконує функцію графічної оболонки, зазвичай знаходиться на 7-му терміналі (TTY7). Це означає, що графічне середовище, таке як GNOME, KDE або інше X-сервер середовище, зазвичай працює на Ctrl + Alt + F7.

Віртуальні термінали (TTY) — це текстові консолі, які дозволяють працювати з командним рядком. Зазвичай за допомогою комбінацій клавіш, таких як Ctrl + Alt + Fn (де n — номер терміналу), користувач може перемикатися між ними.

Графічна оболонка за замовчуванням працює на 7-й консолі, проте це може залежати від конфігурації системи. У деяких сучасних дистрибутивах графічне середовище може запускатися на інших терміналах, наприклад, на TTY1.

Приклад:

Ctrl + Alt + F1-F6 — текстові консолі для роботи з командним рядком.

Ctrl + Alt + F7 — графічне середовище (X-сервер або Wayland).

1. \*\*Чи можлива реєстрація в системі Linux декілька разів під одним і тим же системним ім’ям? Які переваги це може надати?

Механізм реєстрації одного користувача кілька разів:

Користувач може одночасно працювати через кілька віртуальних терміналів (консолей), перемикаючись між ними за допомогою комбінацій клавіш (наприклад, Ctrl + Alt + F1, Ctrl + Alt + F2 тощо). У кожному терміналі буде відкрита нова сесія, але під тим же самим користувачем.

Користувач може одночасно підключатися до однієї і тієї ж машини через SSH (Secure Shell) з різних клієнтських комп'ютерів або через різні сеанси на одному комп'ютері. Це корисно для адміністраторів систем або розробників, які потребують доступу до одного і того ж облікового запису з різних місць.

У графічному середовищі, наприклад GNOME або KDE, користувач може працювати з декількома сесіями, відкривши кілька вікон терміналу або запустивши нові сесії через графічний диспетчер входу (наприклад, через Switch User).

Переваги багаторазової реєстрації під одним ім'ям:

Користувач може працювати над кількома незалежними завданнями одночасно, відкриваючи окремі сесії для кожного з них. Наприклад, в одній сесії можна виконувати компіляцію коду, а в іншій — тестувати програму або редагувати файли.

Адміністратори можуть підключатися до серверів кілька разів, виконуючи одночасно різні дії, такі як встановлення оновлень у одному сеансі та моніторинг системи у іншому. Це підвищує продуктивність та ефективність

Можливість працювати з системою віддалено через SSH дозволяє отримувати доступ до одного і того ж облікового запису з різних локацій, що дуже зручно, коли потрібен багатозадачний доступ до одного й того ж сервера.

Незалежність сесій дозволяє уникнути конфліктів між процесами або завданнями, що виконуються паралельно. Якщо у одній сесії сталася помилка або програма "зависла", це не вплине на інші сесії користувача.

Можна одночасно мати кілька конфігурацій робочого середовища або змінних оточення для різних завдань. Наприклад, різні сесії можуть працювати з різними налаштуваннями проксі або мережевих підключень.