“Київський фаховий коледж зв’язку”

Циклова комісія комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №3**

з дисципліни “Операційні системи”:

**Тема: “Знайомство з базовими командами CLI-режиму в Linux”**

Виконали студенти

групи РПЗ-13А та РПЗ-13Б

Eleven Two Zeroes:

Vlad Sapozhnyk

Max Karpenko

Dmytro Onufriiev

Київ 2024

Робота студентів групи РПЗ-13А та РПЗ-13Б Eleven Two Zeroes: Vlad Sapozhnyk, Max Karpenko and Dmytro Onufriiev.

**Мета роботи:**

1. Знайомство з базовими командами CLI-режиму в Linux.
2. Знайомство з базовими текстовими командами в термінальному режимі роботи в різних ОС.

**Матеріальне забезпечення занять:**

1. ЕОМ типу IBM PC.
2. ОС сімейства Windows (Windows 7).
3. Віртуальна машина – VirtualBox (Oracle).
4. Операційна система GNU/Linux – CentOS.

**Завдання для попередньої підготовки: *Created by Dmytro Onufriiev***  
**Хід роботи. *Created by Vlad Sapozhnyk***

**Контрольнi запитання: *Created by Max Karpenko***

***1. Які типи команд існують в оболонці Bash?***

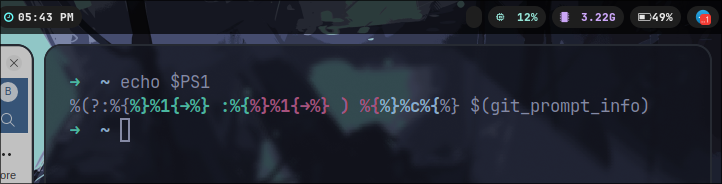
1. **Вбудовані команди оболонки (Builtin commands):** Це команди, які виконуються безпосередньо оболонкою Bash і не вимагають виклику зовнішніх виконуваних файлів. Вони зазвичай забезпечують функціональність, необхідну для управління робочим середовищем оболонки, наприклад, **cd** для зміни директорії, **export** для встановлення значень змінних середовища тощо.
2. **Зовнішні команди:** Це програми, які зберігаються де-інде на файловій системі і викликаються оболонкою Bash для виконання. Ці команди не вбудовані в оболонку і можуть бути частиною операційної системи або встановлені користувачем. Приклади зовнішніх команд включають більшість утиліт Unix/Linux, таких як **ls**, **grep**, **awk** тощо.
3. **Аліаси (Aliases):** Аліаси дозволяють визначати короткі псевдоніми для довгих команд або послідовностей команд, що полегшує їх введення та запам'ятовування. Наприклад, користувач може створити аліас **ll** як короткий спосіб виклику **ls -l**.
4. **Функції оболонки:** Функції в Bash дозволяють групувати кілька команд в один блок, який можна викликати за допомогою одного імені функції. Це спосіб створення складних скриптів та процедур в оболонці, що можна використовувати для автоматизації завдань.
5. **Ключові слова (Keywords):** Це слова, які мають спеціальне значення в Bash. Вони використовуються для управління логікою та потоком виконання скриптів. Приклади ключових слів включають **if**, **else**, **fi**, **for**, **while**, **do**, **done** тощо.

***2. Що таке змінні оточення? Які вони бувають. Як їх можна переглянути в терміналі?***

Змінні оточення - це змінні, які визначають налаштування та параметри робочого середовища операційної системи та програм. Для перегляду змінних оточення в терміналі можна використати команду printenv або env.

**3. Опишіть змінну $PS1. Як в терміналі переглянути її вміст?**

$PS1 - це змінна оточення, яка визначає вигляд командного рядка (prompt) в Bash. Для перегляду її вмісту в терміналі використовуйте команду echo $PS1.



**4. Як можна змінити значення змінної $PS1? Що при цьому відбудеться в рядку запрошенні в bash (рядок запрошення перед початком кожної команди). Як змінити значення цієї змінної не на поточний сеанс, а за замовчуванням?**

Для зміни $PS1 використайте команду export PS1='нове\_значення'. Це змінить вигляд рядка запрошення в Bash на поточний сеанс. Щоб змінити значення не на поточний сеанс, а за замовчуванням, додайте таку ж команду до файлу **~/.bashrc** або **~/.profile** та перезавантажте термінал або виконайте source **~/.bashrc**.

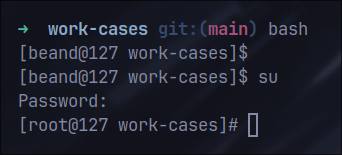
**5. Для чого використовують лапки в оболонці Bash?**

1. Одинарні лапки (' ') зберігають буквальне значення всіх символів в середині.
2. Подвійні лапки (" ") дозволяють інтерпретувати спеціальні символи (наприклад, $, `) в середині.
3. Зворотні лапки (` `) використовують для виконання команд і підстановки результату в місце їх використання.

**6. Для чого використовують інструкції керування, які їх види Ви знаєте?**

1. Умовні оператори (**if, else, elif, case**)
2. Цикли (**for, while, until**)
3. Керування вибором (**select**)
4. Розгалуження (**break, continue**)
5. Повернення значення (**exit, return**)

7. В чому різниця якщо в кінці рядку запрошення bash стоїть символ $ чи #? Наприклад на екрані ми бачимо наступні записи:



1. Символ **$** в кінці рядка запрошення означає, що ви залогінені як звичайний користувач.
2. Символ **#** означає, що ви залогінені як користувач root (суперкористувач).

**8. Яке призначення команд whereis та locate? Яка між ними відмінність?**

**whereis** шукає бінарні файли, вихідні коди та сторінки довідника для програми. locate шукає файли, використовуючи базу даних, створену командою **updatedb**.

Основна відмінність: **whereis** зосереджений на програмному забезпеченні, тоді як **locate** шукає всі файли в системі.

**Висновок:**