**Міністерство освіти і науки України Національний технічний університет**

**«Дніпровська політехніка» Факультет інформаційних технологій Кафедра інформаційних технологій та комп’ютерної інженерії**

**Звіт**

**з лабораторної роботи №6 дисципліни “Операційні системи”**

**Тема роботи: «Створення процесів в GNU/Linux-сумісних ОС»**

Виконав:

студент гр. *шифр групи Ініціали та прізвище студента*

Перевірив:

*посада* каф. ІТКІ *Ініціали та прізвище викладача*

Дніпро 2024

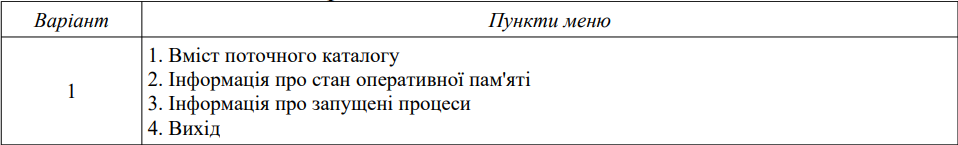
**Лабораторна робота №6**

**Створення процесів в GNU/Linux-сумісних ОС**

**Мета роботи:** вивчити особливості створення процесів в середовищі Linux.

**Хід роботи**

**Завдання**

****

**Код програми-меню**

#include <iostream>

#include <unistd.h>

#include <sys/wait.h>

using namespace std;

void executeCommand(const char\* command)

{

pid\_t pid = fork();

if (pid == 0)

{ // Child process

execlp(command, command, (char\*)NULL); // Execute command

perror("execlp"); // Error handling if exec fails

\_exit(1); // Exit child if exec fails

}

else if (pid > 0)

{ // Parent process

wait(NULL); // Wait for the child process to finish

}

else

{

perror("fork");

exit(1);

}

}

int main()

{

int choice;

do

{

cout << "\nMenu:\n";

cout << "1. Contents of the current directory\n";

cout << "2. Information about the state of the RAM\n";

cout << "3. Information about running processes\n";

cout << "4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice)

{

case 1:

executeCommand("ls");

break;

case 2:

executeCommand("free");

break;

case 3:

executeCommand("ps");

break;

case 4:

cout << "Exiting...\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

}

} while (choice != 4);

return 0;

}

**Результати виконання**

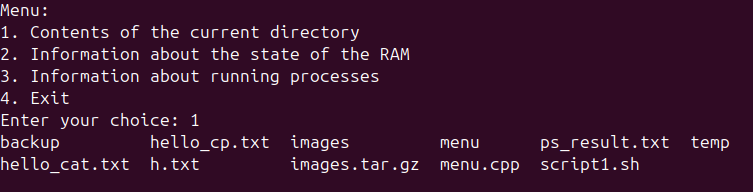
****

Рис. 1. – Виконання пункту меню номер 1

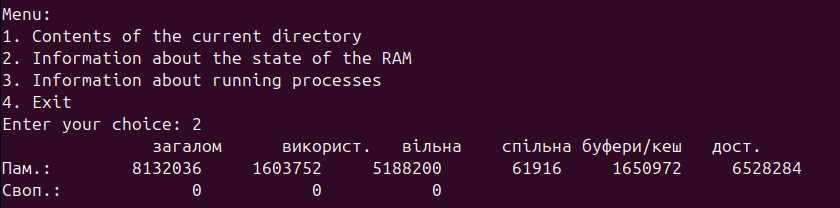


Рис. 2. – Виконання пункту меню номер 2

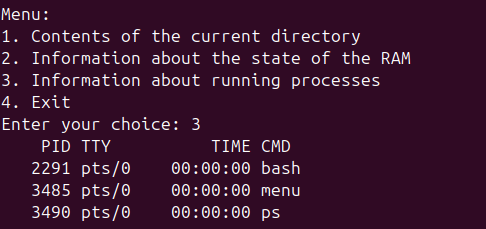


Рис. 3. – Виконання пункту меню номер 3

C++ програма відображає інтерактивне меню, що дозволяє користувачу вибирати системні команди (перегляд файлів, інформацію про пам'ять та процеси). Обрана команда виконується в окремому дочірньому процесі (fork/exec) або безпосередньо (system), забезпечуючи зручний та контрольований спосіб виконання команд в Linux.

**Висновок**

Під час виконання лабораторної роботи створена програма-меню на мові С++, яка надає користувачеві простіший варіант запуску процесів. У даній роботі було успішно розроблено C++ програму з інтерактивним меню для керування системними командами в середовищі Linux. Використання системних викликів fork, execlp та system дозволило реалізувати виконання команд в окремих процесах, забезпечуючи гнучкість та контроль над їх виконанням.