

Міністерство освіти і науки України  
Львівський національний університет імені Івана Франка  
Факультет електроніки

Звіт  
про виконання лабораторної роботи №8  
на тему: «Робота з програмними інтерфейсами  
файлових систем»

Виконав студент  
групи ФЕІ - 23  
факультету електроніки  
Попович Василь  
Викладач:  
Сінькевич О. О.

Львів – 2015

**Мета:** вивести інформацію про ПК, використовуючи інтерфейси файлових систем.

## Завдання:

1. Розробіть програму отримання системної інформації в ОС Linux, використовуючи файлову систему /proc. Програма повинна виводити на екран і записувати у файл інформацію про параметри центрального процесора, стан пам'яті тощо.
2. Розробіть програму отримання системної інформації в ОС Windows XP, використовуючи програмний інтерфейс доступу до системного реєстру. Програма повинна виводити на екран і записувати у файл інформацію про стан і конфігурацію системи.

## Теоретична частина

Більшість UNIX-систем керують різними типами файлових систем із використанням універсального рівня програмного забезпечення, який називають *віртуальною файловою системою* (Virtual File System, VFS). Така система надає прикладним програмам однаковий програмний інтерфейс, що реалізується за допомогою системних викликів роботи з файлами; надає розробникам файлових систем набір функцій, які їм треба реалізувати для інтеграції їхньої системи в інфраструктуру VFS (цей набір функцій називають *інтерфейсом файлової системи*).

Основною метою VFS є забезпечення можливості роботи ОС із максимально широким набором файлових систем.

Рівень VFS забезпечує доступ через стандартні файлові системні виклики до будь-якого рівня програмного забезпечення, що реалізує інтерфейс файлової системи. Програмні модулі, що реалізують інтерфейс файлової системи, називаються *модулями підтримки файлових систем*.

Файлові системи, підтримувані VFS, можуть бути розділені на три основні категорії.

- *Дискові* є файловими системами в їхньому традиційному розумінні (розташовані на диску).
- *Мережні* реалізують прозорий доступ до файлів на інших комп'ютерах через мережу.
- *Спеціальні* або *віртуальні* відображають у вигляді файлової системи те, що насправді файловою системою не є.

Під час розробки VFS були широко використані принципи об'єктної орієнтації. Ця інфраструктура складається із двох основних груп елементів: набору правил, яким мають підлягати файлові об'єкти, і рівня програмного забезпечення для керування цими об'єктами. Базова архітектура VFS визначає три основних об'єктних типи.

- *Об'єкт індексного дескриптора* (inode object) описує набір атрибутів і методів, за допомогою яких відображують файл на рівні файлової системи.
- *Об'єкт відкритого файла* (file object) відображає відкритий файл на рівні процесу.
- *Об'єкт файлової системи* (filesystem object) відображає всю файлову систему; у Linux, його називають *об'єктом суперблока* (superblock object).

## Виконання

### *Завдання:*

Розробив програму отримання системної інформації в ОС Linux, використовуючи файлову систему /proc. Програма виводить на екран інформацію про параметри центрального процесора, стан пам'яті тощо.

### *Код програми:*

```
#include <stdio.h>
#include <string.h>

int main()
{
    FILE* cpu;
    FILE* mem;

    char infoproc[1000];
    size_t bytes_read;
    char* match;
    float speed;

    cpu=fopen("/proc/cpuinfo","r");
    bytes_read=fread(infoproc,1,sizeof(infoproc),cpu);

    fclose(cpu);
    infoproc[bytes_read]='\0';
    printf("\nInformation about CPU\n\n%s\n\n",infoproc);

    char infomemory[500];
    mem=fopen("/proc/meminfo","r");
    bytes_read=fread(infomemory,1,sizeof(infomemory),mem);
    fclose(mem);
```

```

infomemory[bytes_read]=='\0';

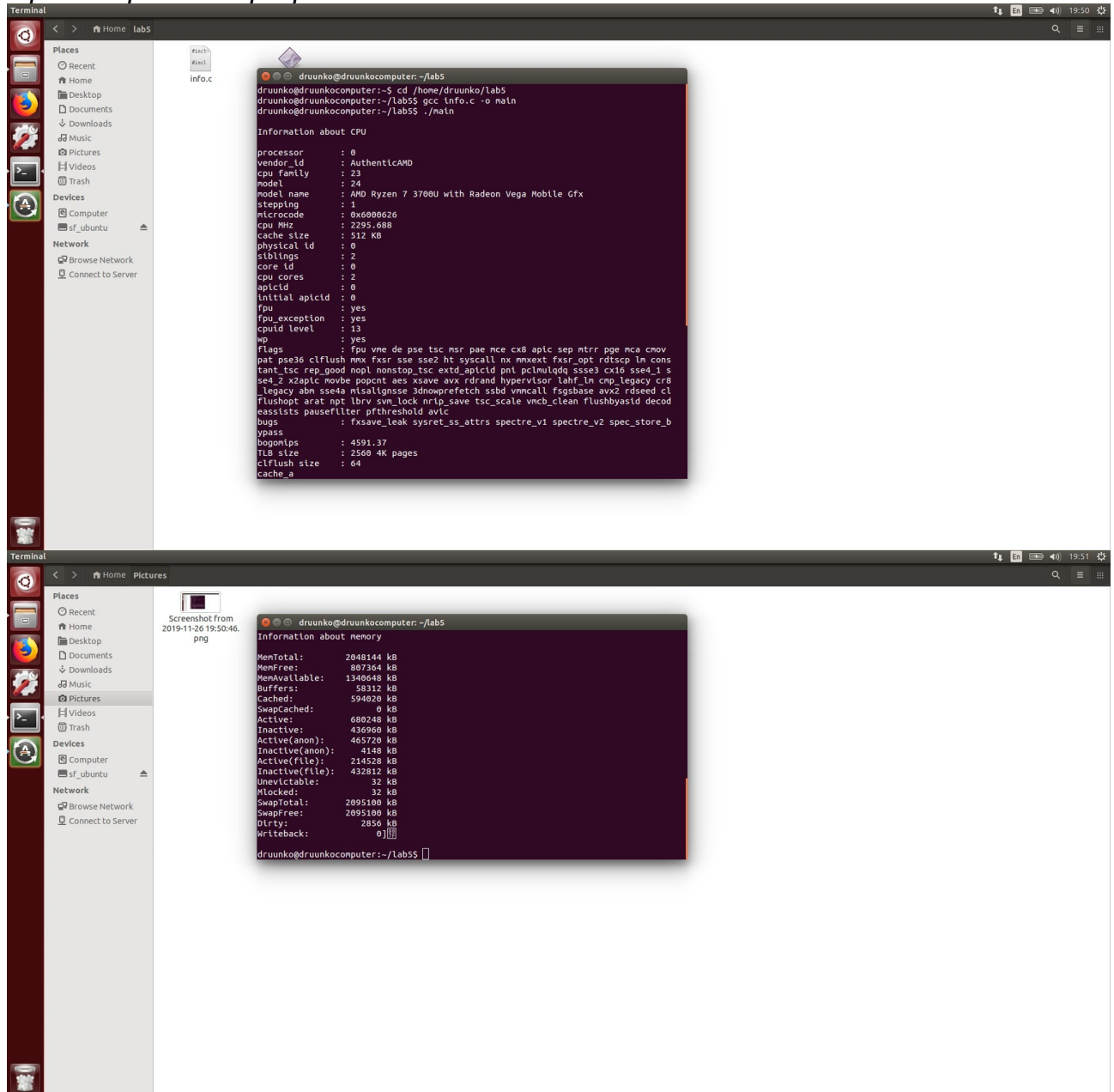
printf("\nInformation about memory\n\n%s\n\n",infomemory);

return 0;

}

```

*Приклад роботи програми:*



**Висновок:** я навчився виводити інформацію про ПК, використовуючи інтерфейси файлових систем.