

Міністерство освіти та науки України
Львівський національний університет ім. Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіоелектронних і комп'ютерних систем

ЗВІТ

про виконання лабораторної роботи №8
«Робота з програмними інтерфейсами файлових
систем»

Виконав:
Студент групи ФЕІ-23
Ангелов М.С.
Перевірив:
ас. Сінькевич О.О.

Львів – 2019

1. Завдання

- 1) Розробити 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 10, використовуючи програмний інтерфейс доступу до системного реєстру. Програма повинна виводити на екран і записувати у файл інформацію про стан і конфігурацію системи.

2. Виконання

2.1. Linux

1) Головний код програми, написаної на C

```
1 #include <stdio.h>
2 #include <sys/vfs.h>
3 #include <stdlib.h>
4 #include <fcntl.h>
5 #include <unistd.h>
6 #include <string.h>
7
8 int main(){
9     FILE * stat = fopen("info.txt", "w");
10    printf("Info from /proc\n");
11    int handle;
12    char minfo[1024+1], *match;
13    ssize_t nread;
14    handle = open("/proc/meminfo", O_RDONLY);
15    if (handle < 0){
16        printf("Failed to read /proc/meminfo\n");
17        fprintf(stat, "Failed to read /proc/meminfo\n");
18    }
19    nread = read(handle, minfo, sizeof(minfo)-1);
20    if (nread < 0){
21        printf("Failed to read from /proc/meminfo\n");
22        fprintf(stat, "Failed to read from /proc/meminfo\n");
23        close(handle);
24    }
25    close(handle);
26    minfo[nread] = '\0';
27    match = minfo;
28    while(*match){
29        char *eol = match + strchr(match, '\n');
30        if (!*eol){
31            break;
32        }
33        *eol++ = 0;
34        char *tag = match;
35        match = eol;
36        char *value = tag + strchr(tag, ':');
37        *value++ = 0;
38        if (strcmp(tag, "MemTotal") == 0){
39            printf("MemTotal: %s\n", value);
40            fprintf(stat, "MemTotal: %s\n", value);
41        }
42        if (strcmp(tag, "MemFree") == 0){
43            printf("MemFree: %s\n", value);
44            fprintf(stat, "MemFree: %s\n", value);
45        }
46        if (strcmp(tag, "MemAvailable") == 0){
47            printf("MemAvailable: %s\n", value);
48            fprintf(stat, "MemAvailable: %s\n", value);
49        }
50        if (strcmp(tag, "Buffers") == 0){
51            printf("Buffers: %s\n", value);
52            fprintf(stat, "Buffers: %s\n", value);
53        }
54        if (strcmp(tag, "Cached") == 0){
55            printf("Cached: %s\n", value);
56            fprintf(stat, "Cached: %s\n", value);
57        }
58    }
59    char cpuinfo[1024];
60    handle = open("/proc/cpuinfo", O_RDONLY);
61    if (handle < 0){
62        printf("Failed to read /proc/cpuinfo\n");
63        fprintf(stat, "Failed to read /proc/cpuinfo\n");
64    }
65    nread = read(handle, cpuinfo, sizeof(cpuinfo)-1);
66    if (nread < 0){
```

2) Виконання з командного рядка.

```
nick@nick:~/info$ gcc info.c -o info
nick@nick:~/info$ ./info
Info from /proc
MemTotal:      4037108 kB
MemFree:       248764 kB
MemAvailable:  1490712 kB
Buffers:       127696 kB
Cached:        1328160 kB
CPU model: Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz
MHz: 3192.000
Cache: 12288 KB
Boost MHz: 6384.00
nick@nick:~/info$ ls
info  info.c  info.txt
```


2.2. Windows

1) Головний код програми, написаної на C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Management;
using Microsoft.Win32;

namespace OSLab7
{
    class Program
    {
        static void Main(string[] args)
        {
            List<string> lines = new List<string>();
            string root = "HKKEY_LOCAL_MACHINE\\HARDWARE\\DESCRIPTION\\System";
            var biosName = Registry.GetValue(root, "SystemBiosVersion", "None");
            lines.Add("Your BIOS info:");
            foreach (string str in (string[])biosName)
            {
                lines.Add(str);
            }
            lines.Add("Your Processor info:");
            int cores = 1;
            var proc0 = Registry.GetValue(root+"\\CentralProcessor\\0", "ProcessorNameString", "None");
            lines.Add(proc0.ToString());
            while (Registry.GetValue(root + "\\CentralProcessor\\"+cores, "ProcessorNameString", null) != null)
            {
                cores++;
            }
            lines.Add("You have "+cores+" virtual cores.");
            ManagementObjectSearcher searcher = new ManagementObjectSearcher("select * from Win32_PhysicalMemory");
            foreach (ManagementObject share in searcher.Get())
            {
                string mfc = "";
                long gb=0, speed = 0;
                foreach (PropertyData pd in share.Properties)
                {
                    if (pd.Name == "ConfiguredClockSpeed")
                        speed = Convert.ToInt32(pd.Value);
                    else if (pd.Name == "Capacity")
                        gb = Convert.ToInt64(pd.Value) / 1024L / 1024 / 1024;
                    else if (pd.Name == "Manufacturer")
                        mfc = (string)pd.Value;
                }
                lines.Add(string.Format("You have a {0} DDR-{1} memory of {2}GB", mfc, speed, gb));
            }
            File.WriteAllLines("result.txt", lines);
            foreach (string line in lines)
            {
                Console.WriteLine(line);
            }
            Console.Read();
        }
    }
}
```

2) Результат виконання .exe

 C:\Users\Niko\source\repos\OSLab7\OSLab7\bin\Debug\OSLab7.exe

```
Your BIOS info:
ALASKA - 1072009
1.60
American Megatrends - 5000D
Your Processor info:
Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz
You have 12 virtual cores.
You have a Kingston DDR-2667 memory of 16GB
You have a Kingston DDR-2667 memory of 16GB
-
```