

Міністерство освіти України  
Харківський національний університет радіоелектроніки

Звіт до лабораторної роботи № 2  
з курсу «Операційні системи UNIX»

Виконав: ст. гр. ПЗПІ-16-3

Губар С.О.

Прийняв: Сокорчук І.П.

Харків 2020

**Мета роботи:** використовуючи Bash доповнити скрипт з першої лабораторної роботи для того, щоб збирати дані про систему користувача.

## **Хід роботи**

У файлі task1.sh з попередньої лабораторної роботи створимо функцію collectInfo, яка буде приймати на вхід ім'я файла, у який потрібно записати усю інформацію. Викличемо її з ім'ям файла, який отримано у ході виконання першої лабораторної роботи:

```
collectInfo() {  
    echo "Date: $(date)" > $1  
    echo "---- Hardware ----" >> $1  
    ....  
}  
collectInfo $outFile  
echo "Created output file $outFile at $(date)."
```

Почнемо збирати інформацію про систему з CPU:

```
cpu=$(cat /proc/cpuinfo | grep 'model name' | uniq)  
  
if [ -z "$cpu" ]; then  
    if [[ $LANG =~ uk_UA ]]; then  
        echo "Немає інформації про CPU" 1>&2  
        cpu="CPU: Невідомо"  
    else  
        echo "Failed to fetch CPU info, skipping..." 1>&2  
        cpu="CPU: Unknown"  
    fi  
else  
    cpu=${cpu##*.}  
    cpu=$(echo $cpu | sed -e 's/^[[:space:]]*//')  
    cpu="CPU: \"$cpu\""  
fi
```

```
echo $cpu >> $1
```

Аналогічно, RAM (код обробки помилок та форматування не наведено тому, що він майже повністю повторює CPU):

```
ram=$(free -m | grep Mem: | awk '{print $2}')
```

Отримаємо інформацію про плату за допомогою dmidecode:

```
manufacturer=$(sudo dmidecode --type baseboard | grep Manufacturer: || echo Unknown)
```

```
echo "Manufacturer:${manufacturer#*:}"
```

```
echo "Manufacturer:${manufacturer#*:}" >> $1
```

```
product_name=$(sudo dmidecode -t baseboard | grep -i 'Product name' || echo Unknown)
```

```
echo "Product:${product_name#*:}"
```

```
echo "Product:${product_name#*:}" >> $1
```

```
serial_number=$(sudo dmidecode -t baseboard | grep -i Serial || echo Unknown)
```

```
echo "System Serial Number:$serial_number"
```

```
echo "System Serial Number:$serial_number" >> $1
```

Інформацію про систему:

```
echo "---- System ----" >> $1
```

```
os_distribution=$(cat /etc/os-release | grep "PRETTY_NAME" | grep -o -E "\.+\\"")
```

```
echo "OS Distribution: $os_distribution"
```

```
echo "OS Distribution: $os_distribution" >> $1
```

```
kernel_version=$(uname -v)
```

```
echo "Kernel version: $kernel_version"
```

```
echo "Kernel version: $kernel_version" >> $1
```

```
created_info=$(sudo dumpe2fs $(mount | grep 'on / ' | awk '{print $1}') | grep 'Filesystem  
created: ')
```

```
echo $created_info
```

```
if [ -z "$created_info" ]; then
```

```
echo "Installation date - error" 1>&2
```

```
created_info="Unknown"
```

```

else
    created_info=${created_info#*:}
    created_info=$(echo $created_info | sed -e 's/^[[:space:]]*//')
fi

```

Та про інтерфейси мережі:

```

net_interfaces=$(ip link show | grep -oE '[0-9]+:\s[^\w]+:' | awk '{print $2}' | sed 's/://'))
if [ -z "$net_interfaces" ]; then
    echo "Network info - error" 1>&2
    echo "\n" >> $1
else
    for var in ${net_interfaces[@]}; do
        ip_addr=$(ip addr show "$var" | grep -E 'inet ' | awk '{print $2}')
        if [ -z "$ip_addr" ]; then
            ip_addr='-/-'
        fi
        echo "${var}: ${ip_addr}" >> $1
    done
fi
echo "----EOF----" >> $1

```

Звернемо увагу, що результат виконання усіх команд було спрямовано у перший аргумент функції, який є файлом, куди ми усе записуємо.

**Висновки:** у ході виконання лабораторної роботи було модифіковано скрипт з першої роботи, який збирає інформацію про систему користувача, вміє обробляти помилки, та враховує локалізацію.