

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Лабораторная работа 1

На тему: «Программа на bash для сбора и
вывода информации о системе»

По дисциплине:
«Администрирование серверов»

Выполнил

Студент 2 курса
Переверзев И. Д.
Гр 221-321
11.04.24

Проверил

Гневшев А. Ю.

МОСКВА 2024

1 ЦЕЛЬ РАБОТЫ

Цель лабораторной работы: получить практические навыки работы с командной оболочкой `bash` и утилитами для получения информации о системе, а также базовых алгоритмических конструкциях скриптов.

2 ПОСТАНОВКА ЗАДАЧИ

Написать скрипт (программу), который собирает информацию о системе и выводит её в интерактивном/автоматическом режимах. Скрипт при запуске без параметров должен показывать пользователю меню для выбора типа информации, которую скрипт может предоставить. Пользователь выбирает пункт меню вводом его номера в строке ввода непосредственно под меню. Последний пункт должен предоставлять возможность выйти из системы. После выбора пункта меню и вывода информации программа должна вывести на экран надпись «Для продолжения нажмите Enter». По нажатию на Enter должно снова отображаться меню. Интерфейс скрипта должен быть на русском языке.

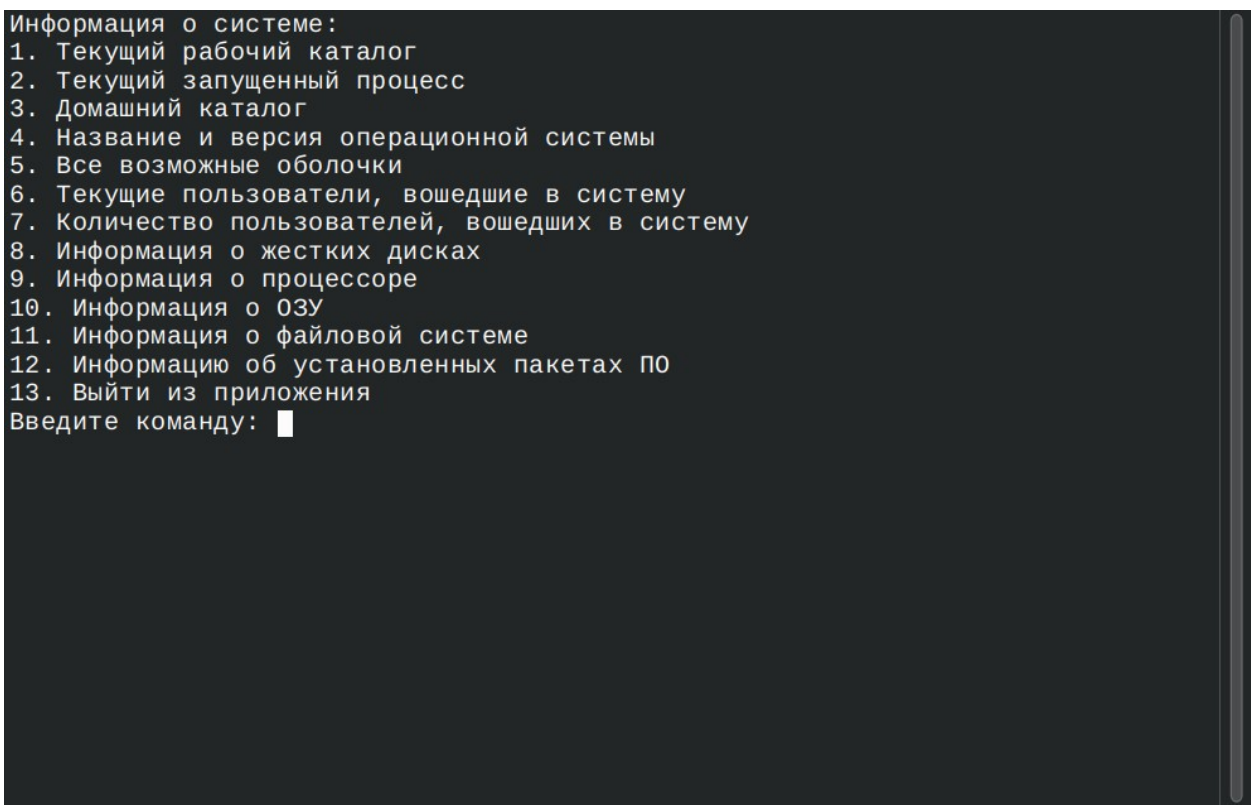
Программа должна собирать и выводить следующую информацию:

- Текущий рабочий каталог
- Текущий запущенный процесс
- Домашний каталог
- Название и версия операционной системы
- Показать все доступные оболочки в вашей системе
- Текущие пользователи, вошедшие в систему
- Количество пользователей, вошедших в систему
- Информация о жестких дисках
- Информация о процессоре
- Информация о памяти
- Информация о файловой системе
- Информация об установленных пакетах ПО

Предусмотреть отдельный пункт меню и опцию командной строки *--tofile <filename>*, по которым программа выводит всю возможную собираемую информацию на экран при выборе этого пункта меню, либо в файл, указанный следующим аргументом после *--tofile* в командной строке. Блоки информации должны разделяться строкой с 40 подряд идущими знаками равенства («=»). В отчёт прикрепить файл output.txt с выводом всей собираемой информации.

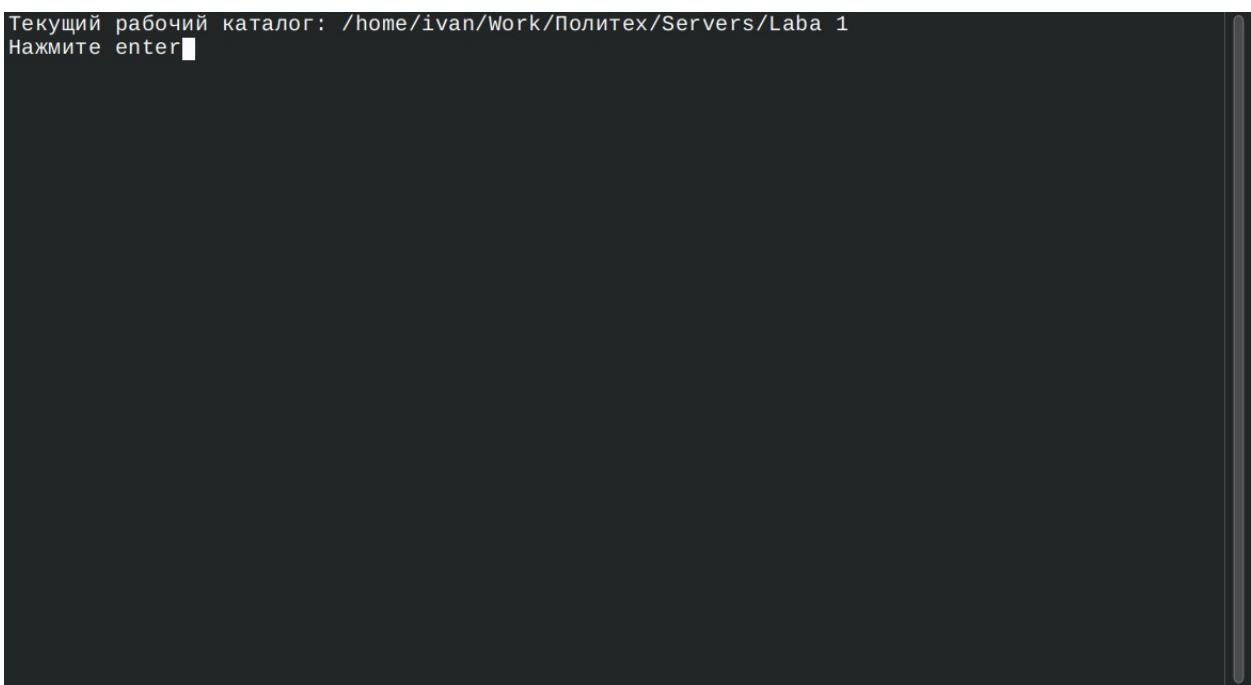
3 ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

3.1 Скриншоты интерфейса программы



```
Информация о системе:  
1. Текущий рабочий каталог  
2. Текущий запущенный процесс  
3. Домашний каталог  
4. Название и версия операционной системы  
5. Все возможные оболочки  
6. Текущие пользователи, вошедшие в систему  
7. Количество пользователей, вошедших в систему  
8. Информация о жестких дисках  
9. Информация о процессоре  
10. Информация о ОЗУ  
11. Информация о файловой системе  
12. Информацию об установленных пакетах ПО  
13. Выйти из приложения  
Введите команду: █
```

Рисунок 1 - Меню выбора действия



```
Текущий рабочий каталог: /home/ivan/Work/Политех/Servers/Laba 1  
Нажмите enter █
```

Рисунок 2 - Текущий рабочий каталог

```
Текущий запущенный процесс:  
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
ivan      41354  0.0  0.0   7660   5492 pts/3    S+   10:19    0:00 /bin/bash -e ./script.s  
Нажмите enter
```

Рисунок 3 - Текущий запущенный процесс

```
Домашний каталог: /home/ivan  
Нажмите enter
```

Рисунок 4 - Домашний каталог пользователя

```
Информация об операционной системе:  
1) Название операционной системы: Arch Linux  
2) Версия ядра: Linux 6.8.4-zen1-1-zen  
Нажмите enter
```

Рисунок 5 - Информация о операционной системе

```
Доступные оболочки:  
/bin/sh  
/bin/bash  
/bin/rbash  
/usr/bin/sh  
/usr/bin/bash  
/usr/bin/rbash  
/usr/bin/git-shell  
/bin/zsh  
/usr/bin/zsh  
Нажмите enter
```

Рисунок 6 - Доступные оболочки терминала

```
Текущие пользователи, вошедшие в систему:  
ivan    tty2      2024-04-12 08:26 (:0)  
ivan    pts/0      2024-04-12 08:26 (:0)  
ivan    pts/2      2024-04-12 09:21 (:0)  
ivan    pts/3      2024-04-12 10:19 (:0)  
Нажмите enter
```

Рисунок 7 - Пользователи, работающие в системе на данный момент

```
На данный момент в системе работает 4 пользователей  
Нажмите enter
```

Рисунок 8 - Количество пользователей, работающих в системе сейчас

```

Информация о диске:
Файловая система  Размер  Использовано  Дост  Использовано%  Смонтировано в
dev                6,8G          0    6,8G           0% /dev
run                6,8G        1,7M    6,8G           1% /run
efivarfs           148K        59K    85K           41% /sys/firmware/efi/efivars
/dev/nvme0n1p2     234G        81G   142G           37% /
tmpfs              6,8G        42M    6,8G           1% /dev/shm
tmpfs              6,8G        22M    6,8G           1% /tmp
/dev/nvme0n1p1     381M       174M   207M           46% /boot
tmpfs              1,4G       104K    1,4G           1% /run/user/1000
Нажмите enter

```

Рисунок 9 - Информация о дисках

```

Информация о процессоре:
Архитектура:          x86_64
CPU op-mode(s):      32-bit, 64-bit
Address sizes:        43 bits physical, 48 bits virtual
Порядок байт:        Little Endian
CPU(s):               8
On-line CPU(s) list: 0-7
ID производителя:     AuthenticAMD
Имя модели:           AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx
Семейство ЦПУ:        23
Модель:               24
Thread(s) per core:   2
Ядер на сокет:         4
Сокетов:              1
Степпинг:             1
Frequency boost:      enabled
CPU(s) scaling MHz:   124%
CPU max MHz:          2100,0000
CPU min MHz:          1400,0000
ВогоMIPS:             4191,88

Виртуализация:        AMD-V
L1d cache:            128 KiB (4 instances)
L1i cache:            256 KiB (4 instances)
L2 cache:             2 MiB (4 instances)
L3 cache:             4 MiB (1 instance)
NUMA node(s):         1
NUMA node0 CPU(s):    0-7
Vulnerability Gather data sampling: Not affected
Vulnerability Itlb multihit:      Not affected
Vulnerability L1tf:               Not affected
Vulnerability Mds:                Not affected
Vulnerability Meltdown:           Not affected
Vulnerability Mmio stale data:     Not affected
Vulnerability Reg file data sampling: Not affected
Vulnerability Retbleed:            Mitigation; untrained return thunk; SMT vulnerable
Vulnerability Spec rstack overflow: Mitigation; Safe RET
Vulnerability Spec store bypass:   Mitigation; Speculative Store Bypass disabled via prctl
Vulnerability Spectre v1:          Mitigation; usercopy/swapgs barriers and __user pointer sanitization
Vulnerability Spectre v2:          Mitigation; Retpolines, IBPB conditional, STIBP disabled, RSB filling, PBRSE-eIBRS Not affected
Vulnerability Srbds:              Not affected
Vulnerability Tsx async abort:     Not affected
Нажмите enter

```

Рисунок 10 - Информация о CPU


```

Информация о памяти:
total      used      free      shared  buff/cache  available
Mem:       13Gi      6,2Gi      4,6Gi      204Mi      3,3Gi       7,4Gi
Swap:      0B        0B        0B
Total:     13Gi      6,2Gi      4,6Gi
Нажмите enter

```

Рисунок 11 - Информация об ОЗУ

```

Информация о файловой системе:
Файловая система Тип      1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
dev               devtmpfs    7103136      0           7103136      0% /dev
run               tmpfs       7123352     1732        7121620      1% /run
efivarfs          efivarfs     148         59          85           41% /sys/firmware/efi/efivars
/dev/nvme0n1p2    ext4        244639028   84151524   147987744     37% /
tmpfs             tmpfs       7123352     42900       7080452      1% /dev/shm
tmpfs             tmpfs       7123356     21800       7101556      1% /tmp
/dev/nvme0n1p1    vfat        389360      177548      211812       46% /boot
tmpfs             tmpfs       1424668     104         1424564      1% /run/user/1000
Нажмите enter

```

Рисунок 12 - Информация о файловой системе

```
Установленные пакеты:
akonadi-import-wizard 24.02.1-1
ark 24.02.1-1
base 3-2
base-devel 1-1
bc 1.07.1-4
bluedevils 1:6.0.3-1
bluez-utils 5.73-4
breeze-gtk 6.0.3-1
breeze-plymouth 6.0.3-1
cantor 24.02.1-1
cervisia 24.02.1-1
chromium 123.0.6312.122-1
colord-kde 24.02.1-1
corectrl 1.4.0-3
dhcpcd 10.0.6-1
discord 0.0.49-1
discover 6.0.3-1
docker 1:26.0.0-1
dolphin 24.02.1-1
dolphin-plugins 24.02.1-1
dqlite 1.16.4-1
drkonqi 6.0.3-1
efibootmgr 18-3
elisa 24.02.1-1
ffmpegthumbs 24.02.1-1
filelight 24.02.1-1
firefox 124.0.2-1
flatpak-kcm 6.0.3-1
gamemode 1.8.1-2
git 2.44.0-1
glad 2.0.5-1
go 2:1.22.2-1
grantlee-editor 24.02.1-1
grub 2:2.12-2
gtop 1.1.5-1
gwenview 24.02.1-1
isoimagewriter 24.02.1-1
kamera 24.02.1-1
kamoso 24.02.1-1
kcharselect 24.02.1-1
kclock 24.02.1-1
kcolorchooser 24.02.1-1
kcron 24.02.1-1
kde-dev-scripts 24.02.1-1
```

Рисунок 13 - Установленные в системе пакеты

3.2 Листинг кода

```
#!/bin/bash -e

# Печатает меню выбора действия
printInfo() {
    echo "Информация о системе:"
    echo "1. Текущий рабочий каталог"
    echo "2. Текущий запущенный процесс"
    echo "3. Домашний каталог"
    echo "4. Название и версия операционной системы"
    echo "5. Все возможные оболочки"
    echo "6. Текущие пользователи, вошедшие в систему"
    echo "7. Количество пользователей, вошедших в систему"
    echo "8. Информация о жестких дисках"
    echo "9. Информация о процессоре"
    echo "10. Информация о ОЗУ"
    echo "11. Информация о файловой системе"
    echo "12. Информацию об установленных пакетах ПО"
    echo "13. Выйти из приложения"
}

# Получение текущего рабочего каталога
showCurrentWorkDirectory() {
    echo "Текущий рабочий каталог: $(pwd)" # present working directory
}

# Получение текущего запущенного процесса
showCurrentProcess() {
    echo "Текущий запущенный процесс:"
    # u - получение процесса для пользователя
    # p - получение процесса по PID
    # $$ - PID текущего процесса
    ps -up $$
}

# Печатает домашний каталог пользователя
showHomeDirectory() {
    echo "Домашний каталог: $HOME"
}

# Печатает информацию об операционной системе: сама ОС и версия ядра
showOSInfo() {
    # hostnamectl
    echo "Информация об операционной системе:"
    echo "1) Название операционной системы: $( hostnamectl | grep
"Operating System:" | sed 's/Operating System:\s//g' )"
    echo "2) Версия ядра: $( hostnamectl | grep "Kernel:" | sed 's/\
s*Kernel:\s//g' )"
}

# Печатает информацию о вошедших пользователях
showCurrentUsers() {
    echo "Текущие пользователи, вошедшие в систему:"
    who
}
```

```

# Печатает возможные оболочки терминала
showAvailableShells() {
    echo "Доступные оболочки:"
    cat /etc/shells | grep '/'
}

# Печатает количество вошедших пользователей
countOfCurrentUsers() {
    echo "На данный момент в системе работает $(who | wc -l)
пользователей"
}

# Печатает информацию о дисках
showDiskInfo() {
    echo "Информация о диске:"
    df -h
}

# Печатает информацию о процессоре
showCpuInfo() {
    echo "Информация о процессоре:"
    lscpu | sed "s/$( lscpu | grep "Флаги" )//g"
}

showFilesystemInfo() {
    echo "Информация о файловой системе:"
    df -T
}

# Печатает информацию об ОЗУ
showMemoryInfo() {
    echo "Информация о памяти: "
    free -ht
}

# Печатает информацию об установленных пакетах
showInstalledPackages() {
    echo "Установленные пакеты:"
    dpkg-query -f='${Package} ${Version} ${Architecture}\n'
}

# Печатает разделитель полей
printSeparator() {
    for (( i=0; i < 100; i++ )); do
        echo -n "="
    done
    echo
}

# Печатает все сведения, которые может собрать программа
showAllInfo() {
    showCurrentWorkDirectory
    printSeparator

    showCurrentProcess

```

```

    printSeparator

    showHomeDirectory
    printSeparator

    showOSInfo
    printSeparator

    showAvailableShells
    printSeparator

    showCurrentUsers
    printSeparator

    countOfCurrentUsers
    printSeparator

    showDiskInfo
    printSeparator

    showCpuInfo
    printSeparator

    showMemoryInfo
    printSeparator

    showFilesystemInfo
    printSeparator

    showInstalledPakages
    printSeparator
}

# Переменная для выбора определенной опции
command=0

# Если передан параметр "--tofile"
if [[ -n "$1" && "$1" == --tofile ]]; then
    if [[ -n "$2" ]]; then
        showAllInfo > $2
        exit
    fi
fi

# До тех пор, пока не будет введена команда выхода
while true; do
    clear
    printInfo
    read -p "Введите команду: " command
    clear
    case $command in
        1) showCurrentWorkDirectory;;
        2) showCurrentProcess;;
        3) showHomeDirectory;;
        4) showOSInfo;;

```

```
5) showAvailableShells;;  
6) showCurrentUsers;;  
7) countOfCurrentUsers;;  
8) showDiskInfo;;  
9) showCpuInfo;;  
10) showMemoryInfo;;  
11) showFilesystemInfo;;  
12) showInstalledPakages;;  
13) exit;;  
esac  
read -p "Нажмите enter"  
done
```