

ЛАБОРАТОРНАЯ РАБОТА 1. ИНТЕРАКТИВНАЯ РАБОТА В СИСТЕМЕ. ПОЛЬЗОВАТЕЛЬСКАЯ УЧЕТНАЯ ЗАПИСЬ .....	2
Упражнение 1.1. Загрузка операционной системы. Вход в систему .....	2
Упражнение 1.2. Виртуальные терминалы .....	2
Упражнение 1.3. Завершение сеанса.....	3
Упражнение 1.4. Команды для получения информации о системе и работающих пользователях.....	3
Упражнение 1.5. Редактирование командной строки командного интерпретатора <i>bash</i> и история команд .....	4
Упражнение 1.6. Свойства пользовательской учетной записи.....	4
Упражнение 1.7. Изменение контекста пользователя.....	5
Упражнение 1.8. Общение между пользователями вычислительной системы .....	6
ЛАБОРАТОРНАЯ РАБОТА 2. РАБОТА СО СПРАВОЧНЫМИ СИСТЕМАМИ.....	7
Упражнение 2.1. Страницы электронного руководства <i>UNIX</i> .....	7
Упражнение 2.2. Встроенная справка командного интерпретатора <i>bash</i> .....	8
Упражнение 2.3. Справочная система <i>GNU Info</i> .....	9
ЛАБОРАТОРНАЯ РАБОТА 3. ПОЛЬЗОВАТЕЛЬСКОЕ ОКРУЖЕНИЕ. ПОЛЬЗОВАТЕЛЬСКИЙ ПРОФИЛЬ.....	10
Упражнение 3.1. Переменные окружения. Переменные командного интерпретатора .....	10
Упражнение 3.2. Конфигурационные файлы пользовательского окружения.	
Пользовательский профиль .....	11
ЛАБОРАТОРНАЯ РАБОТА 4. ФАЙЛЫ И ФАЙЛОВАЯ СИСТЕМА <i>UNIX</i> .....	13
Упражнение 4.1. Изучение структуры файловой системы.....	13
Упражнение 4.2. Типы файлов.....	13
Упражнение 4.3. Манипулирование объектами дерева каталогов <i>UNIX</i> .....	14
Упражнение 4.4. Поиск файлов .....	15
Упражнение 4.5. Операции с файловыми системами.....	16
ЛАБОРАТОРНАЯ РАБОТА 5. ВЛАДЕЛЬЦЫ И ПРАВА ДОСТУПА ОБЪЕКТОВ ФАЙЛОВОЙ СИСТЕМЫ <i>UNIX</i> . .....	17
Упражнение 5.1. Владельцы файлов .....	17
Упражнение 5.2. Права доступа к объектам файловой системы <i>UNIX</i> .....	17
ЛАБОРАТОРНАЯ РАБОТА 6. ОБРАБОТКА ТЕКСТОВОЙ ИНФОРМАЦИИ. УПРАВЛЕНИЕ ВВОДОМ-ВЫВОДОМ КОМАНД.	
Подстановки командного интерпретатора .....	19
Упражнение 6.1. Обработка текстовой информации в <i>UNIX</i> .....	19
Упражнение 6.2. Перенаправление ввода-вывода информации .....	20
Упражнение 6.3. Подстановка вывода команд.....	21
Упражнение 6.4. Подстановка шаблонных символов.....	21
ЛАБОРАТОРНАЯ РАБОТА 7. ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА <i>UNIX</i> .....	23
Упражнение 7.1. Упаковщики и архиваторы <i>UNIX</i> .....	23
Упражнение 7.2. Текстовые редакторы <i>UNIX</i> .....	24
Упражнение 7.3. Файловый менеджер <i>UNIX</i> - <i>Midnight Commander</i> .....	25
Упражнение 7.4. Подсистемы отложенных и периодических заданий. ....	25
ЛАБОРАТОРНАЯ РАБОТА 8. ПРОЦЕССЫ <i>UNIX</i> .....	28
Упражнение 8.1. Типы процессов.....	28
Упражнение 8.2. Атрибуты процессов .....	29
Упражнение 8.3. Задания и сигналы .....	29
ЛАБОРАТОРНАЯ РАБОТА 9. СЕТЕВОЕ ОКРУЖЕНИЕ <i>UNIX</i> .....	31
Упражнение 9.1. Удаленный доступ в систему с помощью протокола <i>SSH</i> .....	31
Упражнение 9.2. Локальная и сетевая почта.....	32
Упражнение 9.3. Передача файлов с помощью протокола <i>FTP</i> .....	32
Упражнение 9.4. Передача файлов с помощью протоколов <i>SCP</i> .....	33
Упражнение 9.5. Синхронизация файлов с помощью протокола <i>RSYNC/SSH</i> .....	33
ЛАБОРАТОРНАЯ РАБОТА 10. СИСТЕМА X <i>WINDOW</i> .....	34
Упражнение 10.1. Создание собственного настольного окружения .....	34
Упражнение 10.2. Настройка настольного окружения <i>Xfce</i> .....	35
Упражнение 10.3. Графический вход в систему.....	35
ЛАБОРАТОРНАЯ РАБОТА 11. ПОДСИСТЕМА ПЕЧАТИ .....	36
Упражнение 11.1. Средства печати <i>UNIX</i> .....	36
Основы ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ КОМАНДНОГО ИНТЕРПРЕТАТОРА. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ.....	38
Создание скелета сценария на языке командного интерпретатора <i>BASH</i> .....	38
Реализация алгоритма сценария на языке командного интерпретатора <i>BASH</i> .....	38

# Лабораторная работа 1. Интерактивная работа в системе. Пользовательская учетная запись

## Упражнение 1.1. Загрузка операционной системы. Вход в систему

1. Включите компьютер.
2. В приглашении загрузчика выбора операционной системы выберите Linux.
3. Проследите за сообщениями системы, выдаваемыми в процессе загрузки. Появление приглашения к вводу имени пользователя, означает окончание процесса загрузки системы. Примечание 1: для отображения полного лога загрузки ОС требуется убрать параметр ядра *quiet*. Примечание 2: если по умолчанию установлен графический вход в ОС, переключитесь на виртуальную консоль сочетанием клавиш "Ctrl-Alt-F1" или "Ctrl-Alt-F2".

```
ayzeka@monadayzek: $ pkill bash
root@monadayzek:# dmesg -T
[BC сен 14 18:07:36 2025] Booting Linux on physical CPU 0x0000000000 [0x610f0000]
[BC сен 14 18:07:36 2025] Linux version 6.12.43+deb13-arm64 (debian-kernel@lists.debian.org) (aarch64-linux-gnu-gcc-14 (Debian 14.2.0-19) 14.2.0, GNU ld (GNU Binutils for Debian) 2.44) #1 SMP Debian 6.12.43-1 (2025-08-27)
[BC сен 14 18:07:36 2025] KASLR disabled due to lack of seed
[BC сен 14 18:07:36 2025] efi: EFI v2.7 by EDK II
[BC сен 14 18:07:36 2025] efi: ACPI 2.0=0x16fc00018 SMBIOS=0xfffffff00 SMBIOS 3.0=0x16fbff000 MOKvar=0x16fbfa0000 INITRD=0x16dfdf1e98 MEMRESERVE=0x16dfdf1f98
[BC сен 14 18:07:36 2025] secureboot: Secure boot disabled
[BC сен 14 18:07:36 2025] APIC: APICv2 support enabled
[BC сен 14 18:07:36 2025] APIC: EDLV support enabled
[BC сен 14 18:07:36 2025] APIC: SGP 0x000000016FC00018 000024 (v02 APPLE )
[BC сен 14 18:07:36 2025] APIC: XSDT 0x000000016FC0FE98 000044 (v01 APPLE Apple V2 00000001 01000013)
[BC сен 14 18:07:36 2025] APIC: FACP 0x000000016FC0FA98 000114 (v06 APPLE Apple V2 00000001 AAPL 20180427)
[BC сен 14 18:07:36 2025] APIC: DSDT 0x000000016FC0F698 000394 (v02 APPLE Apple V2 00000001 AAPL 20180427)
[BC сен 14 18:07:36 2025] APIC: GDTT 0x000000016FC0FC18 000068 (v03 APPLE Apple V2 00000001 AAPL 20180427)
[BC сен 14 18:07:36 2025] APIC: APIC 0x000000016FC0E998 0001AC (v05 APPLE Apple V2 00000001 AAPL 20180427)
[BC сен 14 18:07:36 2025] APIC: MCFG 0x000000016FC0FF98 00003C (v01 APPLE Apple V2 00000001 AAPL 20180427)
[BC сен 14 18:07:36 2025] APIC: Use APIC_SPCR as default console: Yes
```

4. Зафиксируйте приглашение к вводу имени пользователя и расшифруйте его составляющие:

```
root@monadayzek:~# cat /etc/issue
Debian GNU/Linux 13 \n \l

root@monadayzek:~#
```

Так как у меня в эмуляторе шаблон, выводится \h – hostname и \l – tty, то есть нормальный вывод был бы:  
Debian GNU/Linux 13 monadayzek /dev/pts/0

5. В приглашении к вводу имени пользователя введите имя учетной записи *student* или другую доступную учетную запись.

```
root@monadayzek:~# sudo adduser student
Новый пароль:
Повторите ввод нового пароля:
passwd: пароль успешно обновлён
Изменение информации о пользователе student
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
    Полное имя []: student_ayzek
    Номер комнаты []: 10
    Рабочий телефон []: 88005553535
    Домашний телефон []: 88005553535
    Другое []: -
\Is the information correct? [Y/n] y
```

Getent passwd | cut -d: -f1 – выдаст список пользователей где можно найти нового student

```
root@monadayzek:~# getent passwd student
student:x:1001:1001:student_ayzek,10,88005553535,88005553535,-:/home/student:/bin/bash
```

6. Зафиксируйте приглашение к вводу пароля и расшифруйте его составляющие:

```
root@monadayzek:~# passwd -S student
student P 2025-09-14 0 99999 7 -1
```

```
root@monadayzek:~# chage -l student
Последний раз пароль был изменён : сен 14, 2025
Срок действия пароля истекает : никогда
Пароль будет деактивирован через : никогда
Срок действия учётной записи истекает : никогда
Минимальное количество дней между сменой пароля : 0
Максимальное количество дней между сменой пароля : 99999
Количество дней с предупреждением перед деактивацией пароля : 7

student@monadayzek:/root$ su ayzek
Пароль:
```

7. В приглашении к вводу пароля введите *password* или другой установленный пароль. В случае правильного ввода имени пользователя и пароля появится приглашение к вводу команд.

```
root@monadayzek:~# echo "$SHELL"
/bin/bash
root@monadayzek:~# ps -p $$ -o comm=
bash
root@monadayzek:~#
```

8. Зафиксируйте появившиеся сообщения и приглашение к вводу команд и расшифруйте его составляющие:

```
root@monadayzek:~# echo "$PS1"
${debian_chroot:+($debian_chroot)}\u@\h:\w\$
root@monadayzek:~# whoami
root
root@monadayzek:~# hostname
monadayzek
root@monadayzek:~# pwd
/root
root@monadayzek:~# tty
/dev/pts/0
root@monadayzek:~# id -u
0
root@monadayzek:~# id -g
0
```

В echo “\$PS1” - \u – user, \h – host, \w – cwd. Далее что бы понять флаги whoami – даст имя пользователя, hostname – имя хоста, pwd – текущий каталог, tty – устройство терминала, id -u – UID, id -g – GID (\$ – обычный, # – root)

---

---

## Упражнение 1.2. Виртуальные терминалы

1. Используя клавиши **Alt+Fx**, **Alt+←**, **Alt+→** переключитесь и войдите в систему на 2-ом, 4-ом и 6-ом терминале. Примечание: если работаете в графической системе для перехода используйте сочетание **“Ctrl+Alt+Fx”**.

```
ayzek@monadayzek:~$ pkexec bash
root@monadayzek:~# sudo chvt 2
root@monadayzek:~# sudo systemctl start getty@tty4.service
root@monadayzek:~# sudo systemctl start getty@tty6.service
root@monadayzek:~#
```

```
Debian GNU/Linux 13 monadayzek tty2
monadayzek login: ayzek
Password:
Linux monadayzek 6.12.43+deb13-arm64 #1 SMP Debian 6.12.43-1 (2025-08-27) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
ayzek@monadayzek:~$
```

## 2. Зафиксируйте появляющееся сообщение от системы

```
Debian GNU/Linux 13 monadayzek tty6
monadayzek login: ayzek
Password:
Linux monadayzek 6.12.43+deb13-arm64 #1 SMP Debian 6.12.43-1 (2025-08-27) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
ayzek@monadayzek:~$ cat /etc/issue
Debian GNU/Linux 13 \n \l

ayzek@monadayzek:~$ hostname
monadayzek
ayzek@monadayzek:~$ tty
/dev/tty6
ayzek@monadayzek:~$
```

---

---

3. Проследите за появившимися приглашениями к вводу команд.

```
root@monadayzek:~$ echo "$PS1"  
${debian_chroot:+($debian_chroot)}\u@\h:\w\$
```

4. При помощи команды **tty**, получите и зафиксируйте информацию о 2-ом, 4-ом и 6-ом виртуальном терминале соответственно:

```
/dev/tty2  
/dev/tty4  
/dev/tty6
```

5. Вернитесь на терминал #1.

```
/dev/pts/0
```

6. При помощи команды **clear** очистите терминал, проследите за реакцией системы.

Примечание: если данное упражнение невыполнимо с использованием виртуальных терминалов, например, при использовании MacOS X или WSL, то выполните **tty** для эмулятора терминала.

```
root@monadayzek:~#
```

## Упражнение 1.3. Завершение сеанса

1. Последовательно переключаясь между виртуальными терминалами, завершите сеанс работы с системой на терминалах, на которых вы входили в систему, при помощи команд **exit**, **logout**, либо используя комбинацию клавиш **^D**.

```
Debian GNU/Linux 13 monadayzek tty6
```

```
monadayzek login:
```

2. Проследите за соответствующей реакцией системы.

- Дает повторный ввод логина и пароля

## Упражнение 1.4. Команды для получения информации о системе и работающих пользователях

При помощи следующих команд: **whoami**, **id**, **users**, **who**, **w**, **date**, **cal**, **uname**, **uptime** получите информацию о системе и пользователях и зафиксируйте ее:

```
root@monadayzek:~# whoami
root
root@monadayzek:~# id
uid=0(root) gid=0(root) группы=0(root)
root@monadayzek:~# users
ayzek ayzek ayzek ayzek ayzek
root@monadayzek:~# who
ayzek    seat0      2025-09-14 18:58
ayzek    tty3       2025-09-14 18:58
ayzek    seat0      2025-09-14 19:09
ayzek    tty6       2025-09-14 19:09
ayzek    seat0      2025-09-14 19:01
ayzek    tty2       2025-09-14 19:01
root@monadayzek:~# w
 19:09:34 up 1:01, 3 users,  load average: 0,08, 0,08, 0,05
USER   TTY     FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
ayzek  tty3    -           18:58    1:01m  0.00s ? /usr/libexec/gnome-session-binary
ayzek  tty6    -           19:09   22.00s  0.01s 0.01s -bash
ayzek  tty2    -           19:01    7:58   0.02s 0.02s -bash
root@monadayzek:~# date
Вс 14 сен 2025 19:09:36 MSK
root@monadayzek:~# cal
bash: cal: команда не найдена
root@monadayzek:~# uname
Linux
root@monadayzek:~# uptime
 19:09:45 up 1:02, 3 users,  load average: 0,06, 0,07, 0,05
root@monadayzek:~#
```

а. имя текущего пользователя:

root

---

---

б. его идентификаторы UID, GID и идентификаторы вторичных групп:

0 так как root (#)

---

---

в. количество и имена пользователей, работающих в системе, их терминалы, время работы в системе и время входа в систему, выполняемые команды:

3 пользователя Ayzek – tty 3,6,2 + время на скриншоте

---

---

d. системная дата и время:

по МСК – на скриншоте

---

e. календарь текущего месяца:

```
monadayzek@MacBook-Air-Monad ~ % cal
    September 2025
Su Mo Tu We Th Fr Sa
      1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30
```

команды cal нет , но на macOS появляется календарь с выделенной указанной датой

f. версия операционной системы компьютера (uname ???):

```
ayzek@monadayzek:~$ uname -a
Linux monadayzek 6.12.43+deb13-arm64 #1 SMP Debian 6.12.43-1 (2025-08-27) aarch
4 GNU/Linux
```

ZSH - monadayzek@MacBook-Air-Monad ~ % uname -a
Darwin MacBook-Air-Monad.local 24.6.0 Darwin Kernel Version 24.6.0:
Mon Jul 14 11:30:51 PDT 2025; root:xnu-
11417.140.69~1RELEASE\_ARM64\_T8112 arm64

---

Linux –

---

g. время работы операционной системы от момента запуска и нагрузку на операционную систему:

19:09:45 + load avg = 0.06, 0.07, 0.05

---

---

## Упражнение 1.5. Редактирование командной строки командного интерпретатора bash и история команд

- Используя клавиши **↑**, **↓** просмотрите историю вводимых ранее команд.
- Используя клавиши **←**, **→**, **Del**, **BackSpace**, **^H**, **^?**, **^U** потренируйтесь редактировать вводимую команду.

3. Начните набирать любую известную вам команду (первые несколько букв), воспользуйтесь клавишей **Tab** для автоматического завершения ввода команды.
4. При помощи команды: **history** получите историю ранее вводимых команд и зафиксируйте три последние:

```
root@monadayzek:~# history
 1 dmesg -T
 2 dmesg -T
 3 dmesg -T
 4 cat /etc/issue
 5 login
 6 sudo systemctl start getty@tty2.service
 7 echo "$PS1"
 8 tty
 9 cat etc/issue
10 cat /etc/issue/
11 cat /etc/issue
12 hostname
13 tty
14 getent passwd student
15 getent passwd student
16 getent passwd | cut -d: -f1
17 getent passwd ayzek
18 passwd -S student
19 passwd -S ayzek
20 sudo adduser student
21 getent passwd student
22 passwd -S student
23 change -l student
24 chage -l student
25 echo "$SHELL"
26 ps -p $$ -o comm=
27 echo
28 echo "$PS1"
29 whoami
30 hostname
31 pwd
32 tty
33 id -u
34 id -g
35 su student
36* sudo chvt
37 sudo systemctl start getty@tty4.service
38 sudo systemctl start getty@tty6.service
39 sudo chvt 6
40 clear
41 tty
42 sudo chvt 6
43 whoami
44 id
45 users
46 who
47 w
48 date
49 cal
50 uname
51 uptime
52 history
```

5. Повторите любую ранее введенную команду при помощи ссылки на ее номер: **!n**, где **n** - номер введенной ранее команды.

```
root@monadayzek:~# !47
w
 19:15:20 up 1:07, 3 users, load average: 0,02, 0,05, 0,04
USER   TTY      FROM           LOGIN@    IDLE    JCPU    PCPU  WHAT
ayzek  tty3     -          18:58    1:07m  0.00s  ?  /usr/libexec/gnome-session-binary
ayzek  tty6     -          19:09    6:08   0.01s  0.01s -bash
ayzek  tty2     -          19:01   13:44   0.02s  0.02s -bash
root@monadayzek:~#
```

## Упражнение 1.6. Свойства пользовательской учетной записи

- При помощи команды: **passwd** измените пароль учетной записи. Зафиксируйте появляющиеся сообщения:

```
ayzek  tty2      19:01
root@monadayzek:~# passwd
Новый пароль:
Повторите ввод нового пароля:
passwd: пароль успешно обновлён
```

- 
- 
- 
2. При помощи команды: **finger** посмотрите и зафиксируйте информацию о свойствах учетных записей student и суперпользователя соответственно:

```
root@monadayzek:~# getent passwd student
student:x:1001:1001:student_ayzek,10,88005553535,88005553535,-:/home/student:/bin/bash
root@monadayzek:~# getent passwd root
root:x:0:0:root:/root:/bin/bash
root@monadayzek:~#
```

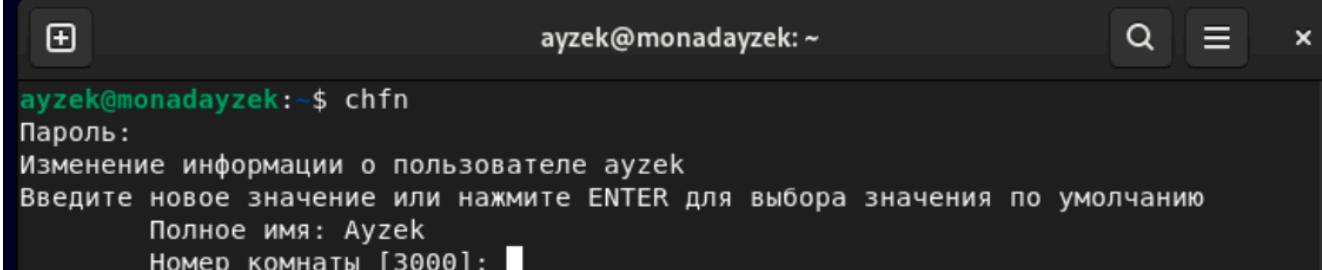
Так как в 13 debian не встроена finger – делаю через getent которая выдает имя:хэш Login/UID/GID/GECOS (ФИО)/ home/ shell

---

---

3. При помощи команды: **chfn** измените содержание информационного свойства (GECOS) учетной записи student. Зафиксируйте появляющиеся сообщения:

```
ayzek@monadayzek:~$ chfn
Пароль:
Изменение информации о пользователе ayzek
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
Полное имя: Ayzek
Номер комнаты []: 3000
Рабочий телефон []: 88005553535
Домашний телефон []: 8800555342
```



```
ayzek@monadayzek:~$ chfn
Пароль:
Изменение информации о пользователе ayzek
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
Полное имя: Ayzek
Номер комнаты [3000]:
```

```
root@monadayzek:~# chfn student
Изменение информации о пользователе student
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
Полное имя [student_ayzek]: haskell
Номер комнаты [10]: 33
Рабочий телефон [88005553535]: 99999999
Домашний телефон [88005553535]: 993939349
Другое [-]: 3939
```

- 
- 
- 
4. При помощи команды: **chsh** измените начальный командный интерпретатор учетной записи student. Зафиксируйте появляющиеся сообщения:

```
root@monadayzek:~# sudo chsh -s /bin/bash student
root@monadayzek:~# echo $?
0
```

```
ayzek@monadayzek:~$ chsh
Пароль:
Изменение командной оболочки для ayzek
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
Командная оболочка [/bin/bash]:
```

```
ayzek@monadayzek:~$ chsh
Пароль:
Изменение командной оболочки для ayzek
Введите новое значение или нажмите ENTER для выбора значения по умолчанию
Командная оболочка [/bin/bash]: /bin/sh
```

Chsh ничего не выдает при успехе но можно узнать об успехе с помощью echo \$? Вывел 0 – успех

- Посмотрите и зафиксируйте изменения в свойствах учетной записи student(/etc/passwd):

```
root@monadayzek:~# getent passwd student
student:x:1001:1001:haskell,33,99999999,993939349,3939:/home/student:/bin/bash
ayzek@monadayzek:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
dhpcd:x:100:65534:DHCP Client Daemon:/usr/lib/dhpcd:/bin/false
tss:x:101:103:TPM software stack:/var/lib/tpm:/bin/false
systemd-timesync:x:991:991:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:990:990:System Message Bus:/nonexistent:/usr/sbin/nologin
sshd:x:989:65534:sshd user:/run/sshd:/usr/sbin/nologin
dnsmasq:x:999:65534:dnsmasq:/var/lib/misc:/usr/sbin/nologin
avahi:x:102:107:Avahi mDNS daemon:/run/avahi-daemon:/usr/sbin/nologin
speech-dispatcher:x:103:29:Speech Dispatcher:/run/speech-dispatcher:/bin/false
usbmux:x:104:46:usbmux daemon:/var/lib/usbmux:/usr/sbin/nologin
cups-pk-helper:x:105:108:user for cups-pk-helper service:/nonexistent:/usr/sbin/nologin
fwupd-refresh:x:988:988:Firmware update daemon:/var/lib/fwupd:/usr/sbin/nologin
geoclue:x:106:110::/var/lib/geoclue:/usr/sbin/nologin
gnome-remote-desktop:x:987:987:GNOME Remote Desktop:/var/lib/gnome-remote-desktop:/usr/sbin/nologin
saned:x:107:111::/var/lib/saned:/usr/sbin/nologin
polkitd:x:986:986:User for polkitd:/:/usr/sbin/nologin
rtkit:x:108:112:RealtimeKit:/proc:/usr/sbin/nologin
colord:x:109:113:colord colour management daemon:/var/lib/colord:/usr/sbin/nologin
Debian-gdm:x:110:114:Gnome Display Manager:/var/lib/gdm3:/bin/false
ayzek:x:1000:1000:Ayzek,3000,8800555353,8800555342:/home/ayzek:/bin/sh
student:x:1001:1001:haskell,33,99999999,993939349,3939:/home/student:/bin/bash
vinnie:x:1002:1002:VinniePuh,111,8888888,3333333,113:/home/vinnie:/bin/bash
```

---

Можно сделать cat /etc/passwd

---

6. Завершите сеанс работы пользователя в системе и заново войдите в систему, проследите за изменениями, произошедшими вследствие смены начального командного интерпретатора:

```
root@monadayzek:~# exit  
exit
```

---

## Упражнение 1.7. Изменение контекста пользователя

1. При помощи команды: **su** измените контекст текущего пользователя *student* на контекст пользователя *vinnie* (или другого пользователя), и зафиксируйте его идентификаторы UID, GID и идентификаторы вторичных групп. Примечание: если второго пользователя в системе не существует, то добавьте его с помощью команд **useradd** и **passwd**:

```
root@monadayzek:~# adduser vinnie  
Новый пароль:  
Повторите ввод нового пароля:  
passwd: пароль успешно обновлён  
Изменение информации о пользователе vinnie  
Введите новое значение или нажмите ENTER для выбора значения по умолчанию  
    Полное имя []: VinniePuh  
    Номер комнаты []: 111  
    Рабочий телефон []: 88888888  
    Домашний телефон []: 33333333  
    Другое []: 113  
Is the information correct? [Y/n] y  
root@monadayzek:~# su vinnie  
vinnie@monadayzek:/root$  
  
vinnie@monadayzek:/root$ id  
uid=1002(vinnie) gid=1002(vinnie) группы=1002(vinnie),100(users)  
vinnie@monadayzek:/root$
```

- 
- 
2. При помощи команды: **exit** вернитесь в контекст текущего пользователя *student*, убедитесь в этом, проверив его идентификаторы UID, GID и идентификаторы вторичных групп:

```
root@monadayzek:~# su vinnie  
vinnie@monadayzek:/root$ exit  
exit  
root@monadayzek:~#
```

```
root@monadayzek:~# id  
uid=0(root) gid=0(root) группы=0(root)  
root@monadayzek:~#
```

---

## Упражнение 1.8. Общение между пользователями вычислительной системы

1. Переключитесь и войдите в систему на свободном виртуальном терминале, воспользовавшись учетной записью *vinnie*. Вернитесь на терминал пользователя *student*.
2. При помощи команды **write** пошлите различные сообщения (завершая ввод каждого сообщения управляющим символом **^D** - символом завершения ввода) пользователю *vinnie*, проследите за появляющимися на терминале пользователя *vinnie* сообщениями:

```
root@monadayzek:~# sudo sh -c  
root@monadayzek:~# write vinnie tty6  
bash: write: команда не найдена  
root@monadayzek:~# sudo apt-get install -y bsdextrautils  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Уже установлен пакет bsdextrautils самой новой версии (2.41-5).  
bsdextrautils помечен как установленный вручную.  
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.  
root@monadayzek:~#
```

```
ayz@monadayzek:~$ su vinnie  
Пароль:  
vinnie@monadayzek:/home/ayz$ who -u  
ayz      seat0    2025-09-14 18:58 ?          3585  
ayz      tty3     2025-09-14 18:58 дав      3585  
ayz      seat0    2025-09-14 19:09 ?          5616  
ayz      tty6     2025-09-14 19:09 .          5616  
vinnie   pts/0    2025-09-14 19:29 .          6206  
ayz      seat0    2025-09-14 19:01 ?          4653  
ayz      tty2     2025-09-14 19:01 00:28    4653  
vinnie@monadayzek:/home/ayz$ hello vinka  
puh
```

Так как write не найдена устанавливаю с apt-get  
Не устанавливается (write) сделал через tee

Echo -e "hello vinka \n puh" | sudo tee dev/tty6

После ctrl + D для EOF из сообщения

- 
3. При помощи команды **mesg** отключите возможность приема сообщений пользователем *vinnie*. Попробуйте послать сообщения пользователю *vinnie*, проследите за появляющимися сообщениями:

Можно на терминале *vinnie* (tty6) запретить прием сообщений вместо mesg n – chmod o-w “\$(tty)”

После echo “you have no rights vinnie” > /dev/tty6 – permission denied  
Вернуть можно с chmod o+w “\$tty”

4. При помощи команды **wall** пошлите сообщение всем пользователям сразу (завершая ввод каждого сообщения управляемым символом **^D** - символом завершения ввода), проследите за появляющимися сообщениями:

```
root@monadazek:~# wall
I use Haskell BTW

Broadcast message from root@monadazek (pts/0) (Sun Sep 14 19:47:52 2025):
I use Haskell BTW

Ha tty6 (vinnie) :
Broadcast message from root@monadazek (pts/0) (Sun Sep 14 19:47:52 2025):
I use Haskell BTW
```

---

---

Лабораторная работа зачтена:  
Дата:  
Подпись преподавателя:

# Лабораторная работа 2. Работа со справочными системами

## Упражнение 2.1. Страницы электронного руководства UNIX

- При помощи команды: **man** познакомьтесь со справкой по команде **man**.

Зафиксируйте назначение команды **man**:

```
MAN(1)                               General Commands Manual      MAN(1)

NAME
  man, apropos, whatis - display online manual documentation pages

SYNOPSIS
  man [-adho] [-t | -w] [-M manpath] [-P pager] [-S mansect]
  [-m arch[:machine]] [-p [eprtv]] [mansect] page ...
  man -f [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
  whatis [-d] [-s mansect] keyword ...
  man -k [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
  apropos [-d] [-s mansect] keyword ...

DESCRIPTION
  The man utility finds and displays online manual documentation pages. If
  mansect is provided, man restricts the search to the specific section of
  the manual.

  The sections of the manual are:
    1. General Commands Manual
    2. System Calls Manual
  :
```

С помощью команды `run-help man`, в zsh

---

---

- Страницы руководства выводятся на терминал постранично при помощи программы постраничной разбивки (PAGER). Вызовите встроенную справку по программе постраничной разбивки с помощью клавиши **h**. Ознакомьтесь с основами работы со справочной системой и зафиксируйте основные комбинации клавиш, используемые для навигации и поиска в справочной системе, а именно:

- клавиши построчного и постраничного листания вверх и вниз:

```
SYNOPSIS
  man [-adho] [-t | -w] [-M manpath] [-P pager] [-S mansect]
  [-m arch[:machine]] [-p [eprtv]] [mansect] page ...
  man -f [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
  whatis [-d] [-s mansect] keyword ...
  man -k [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
  apropos [-d] [-s mansect] keyword ...

DESCRIPTION
  The man utility finds and displays online manual documentation pages.
  mansect is provided, man restricts the search to the specific section
  of the manual.

  The sections of the manual are:
    1. General Commands Manual
    2. System Calls Manual
    3. Library Functions Manual
    4. Kernel Interfaces Manual
    5. File Formats Manual
    6. Games Manual
    7. Miscellaneous Information Manual
  :
```

b. клавишу выхода из справочной системы: / :q

```
monadayzek@MacBook-Air-Monad ~ % run-help man
monadayzek@MacBook-Air-Monad ~ %
```

---

---

---

c. клавиши поиска информации в странице руководства:

```
-o      Force use of non-localized manual pages. See IMPLEMENTATION NOTES for how locale specific searches work. Overrides the LC_ALL, LC_CTYPE, and LANG environment variables.
-p [eprtv]
      Use the list of given preprocessors before running nroff(1) or troff(1). Valid preprocessors arguments:
      e      eqn(1)
```

/pattern, ?pattern :

---

---

---

d. клавиши повторения предыдущего поиска: ->

/ = поиск по строке вперед  
n – повторить поиск вперед  
N – повторить поиск назад

---

---

---

3. При помощи клавиш поиска и повторения поиска найдите на текущей странице все появления слова **page**, зафиксируйте использованные сочетания клавиш:

```
man, apropos, whatis - display online manual documentation pages

SYNOPSIS
man [-adho] [-t | -w] [-M manpath] [-P pager] [-S mansect] [-m arch[:machine]] [-p [eprtv]] [mansect]
      page ...

man -f [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
whatis [-d] [-s mansect] keyword ...

man -k [-d] [-M manpath] [-P pager] [-S mansect] keyword ...
apropos [-d] [-s mansect] keyword ...

DESCRIPTION
The man utility finds and displays online manual documentation pages. If mansect is provided, man
restricts the search to the specific section of the manual.

The sections of the manual are:
1. General Commands Manual
2. System Calls Manual
3. Library Functions Manual
4. Kernel Interfaces Manual
5. File Formats Manual
6. Games Manual
7. Miscellaneous Information Manual
8. System Manager's Manual
9. Kernel Developer's Manual

Options that man understands:

-M manpath
Forces a specific colon separated manual path instead of the default search path. See
manpath(1). Overrides the MANPATH environment variable.

-P pager
Use specified pager. Defaults to "less -sR" if color support is enabled, or "less -s".
Overrides the MANPAGER environment variable, which in turn overrides the PAGER environment
variable.
```

:

/page (предыдущее N | след. n)

4. При помощи команды: **man <номер раздела> intro** ознакомьтесь с назначением всех восьми разделов справочной системы. Зафиксируйте их назначение:

<b>INTRO(1)</b>	<b>General Commands Manual</b>	<b>INTRO(1)</b>
-----------------	--------------------------------	-----------------

**NAME**  
**intro** - introduction to general commands (tools and utilities)

**DESCRIPTION**  
Section one of the manual contains most of the commands which comprise the BSD user environment. Some of the commands included in section one are text editors, command shell interpreters, searching and sorting tools, file manipulation commands system status commands, remote file copy commands, mail commands, compilers and compiler tools, formatted output tools, and line printer commands.

All commands set a status value upon exit which may be tested to see if the command completed normally. The exit values and their meanings are explained in the individual manuals. Traditionally, the value 0 signifies successful completion of the command.

**SEE ALSO**  
**man(1), intro(2), intro(4), intro(8),**

Tutorials in the [UNIX User's Manual Supplementary Documents](#).

```

PASSWD(1)                                General Commands Manual                               PASSWD(1)

NAME
    passwd - modify a user's password

SYNOPSIS
    passwd [-i infosystem [-l location]] [-u authname] [user]

DESCRIPTION
    The passwd utility changes the user's password. If the user is not the super-user, passwd first prompts
    for the current password and will not continue unless the correct password is entered.

    When entering the new password, the characters entered do not echo, in order to avoid the password being
    seen by a passer-by. The passwd utility prompts for the new password twice in order to detect typing
    errors.

    The new password should be at least six characters long and not purely alphabetic. Its total length
    should be less than _PASSWORD_LEN (currently 128 characters), although some directory systems allow
    longer passwords. Numbers, upper case letters, and meta characters are encouraged.

    Once the password has been verified, passwd communicates the new password to the directory system.

-i infosystem
    This option specifies where the password update should be applied. Under Mac OS X 10.5 and later,
    supported directory systems are:

        PAM      (default) Pluggable Authentication Modules.

        opendirectory
            A system conforming to Open Directory APIs and supporting updates (including LDAP, etc). If
            no -l option is specified, the search node is used.

        file    The local flat-files (included for legacy configurations).

        nis     A remote NIS server containing the user's password.

-l location

```

Допустим раздел – passwd, open, read, printf, /dev/\*, man, games, ascii, ifconfig

---



---

5. При помощи команды: **whatis** получите краткую справку по всем известным вам командам. Зафиксируйте и расшифруйте составляющие нескольких описаний команд:

**whatis ls lsof neofetch htop**

```

ls(1)                                     - list directory contents
lsof(8)                                    - list open files
neofetch(1)                                 - A fast, highly customizable system info script
htop(1), pcp-htop(1)                      - interactive process viewer

```

6. При помощи команды: **apropos** воспользуйтесь контекстным поиском справочной информации по любому ключевому слову. Зафиксируйте и расшифруйте составляющие нескольких найденных ссылок:

Поиск по контексту например сору даст cp :

```

cp(ops(n), transfer::copy(n) - Data transfer foundation
cp(1)                                         - copy files
cpio(1)                                       - copy files to and from archives
dd(1)                                         - convert and copy a file
ditto(1)                                       - copy directory hierarchies, create and extract archives

```

---



---

## Упражнение 2.2. Встроенная справка командного интерпретатора bash

1. При помощи команды: **help** help ознакомьтесь со справкой по команде **help**.

Зафиксируйте назначение команды **help**:

Сначала переключусь с zsh на bash (так как в макос по стандарту zsh) : bash  
затем help:

```
monadayzek@MacBook-Air-Monad ~ % bash
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
bash-3.2$ help help
help: help [-s] [pattern ...]
      Display helpful information about builtin commands. If PATTERN is
      specified, gives detailed help on all commands matching PATTERN,
      otherwise a list of the builtins is printed. The -s option
      restricts the output for each builtin command matching PATTERN to
      a short usage synopsis.
bash-3.2$ help
GNU bash, version 3.2.57(1)-release (arm64-apple-darwin25)
These shell commands are defined internally. Type 'help' to see this list.
Type 'help name' to find out more about the function `name'.
Use 'info bash' to find out more about the shell in general.
Use 'man -k' or 'info' to find out more about commands not in this list.

A star (*) next to a name means that the command is disabled.

JOB_SPEC [&]                      (( expression ))
. filename [arguments]           :
[ arg... ]                         [[ expression ]]
alias [-p] [name[=value] ...]     bg [job_spec ...]
bind [-lpvSPVS] [-m keymap] [-f fi break [n]
builtin [shell-builtin [arg ...]] caller [EXPR]
case WORD in [PATTERN [| PATTERN]...]. cd [-L|-P] [dir]
command [-pVv] command [arg ...] compgen [-abcdefgjksuv] [-o option
complete [-abcdefgjksuv] [-pr] [-o continue [n]
declare [-afFirtx] [-p] [name[=val dirs [-clpv] [+N] [-N]
disown [-h] [-arl] [jobspec ...] echo [-neE] [arg ...]
enable [-pnds] [-a] [-f filename] eval [arg ...]
exec [-cl] [-a name] file [redirec exit [n]
export [-nf] [name[=value] ...] or false
fc [-e ename] [-nlr] [first] [last fg [job_spec]
```

---

---

---

---

---

---

2. При помощи команды: **help** выведите список встроенных команд **bash**.

По стандартным –compgen -b

3. Посмотрите встроенную справку по командам: **cd**, **history**, **logout**, **pwd**. Зафиксируйте назначение этих команд:

help -b (-b – команды) (cd – в директорию, history – история zsh, logout – выйти с аккаунта, pwd – абсолютный путь)

---

---

---

## Упражнение 2.3. Справочная система GNU Info

- При помощи команды: **info info** ознакомьтесь со справкой по команде **info**. Примечание: в системах отличных от GNU данная справочная система может отсутствовать.

Выход из bash – exit – перейду в zhs

в макос нет info но есть texinfo (можно скачать МП homebrew)

- brew install texinfo

Зафиксируйте назначение команды **info**:

```
Text: Stand-alone Info, Up: (dir)
Stand-alone GNU Info
*****
This documentation describes the stand-alone Info reader which you can
use to read Info documentation.

If you are new to the Info reader, then you can get started by typing
'H' for a list of basic key bindings. You can read through the rest of
this manual by typing <SPC> and <DEL> (or <Space> and <Backspace>) to
move forwards and backwards in it.

* Menu:
* Stand-alone Info:: What is Info?
* Node Commands:: Commands for selecting a new node.
* More Node Commands:: Further commands for changing the node.
* Cursor Commands:: Commands which move the cursor within a node.
* Scrolling Commands:: Commands for reading the text within a node.
* Searching Commands:: Commands for searching an Info file.
* Index Commands:: Commands for looking up in indices.
* Xref Commands:: Commands for selecting cross-references.
* Window Commands:: Commands which manipulate multiple windows.
* Printing Nodes:: How to print out the contents of a node.
* Miscellaneous Commands:: A few commands that defy categorization.
* Invoking Info:: Options you can pass on the command line.
* Variables:: How to change the default behavior of Info.
* Colors and Styles:: Customize the colors used by Info.
* Custom Key Bindings:: How to define your own key-to-command bindings.
* MS-DOS/Windows Keybindings:: Alternate DOS/Windows keybindings.

Appendices
* Index:: Global index.

-----Info: (info-stnd)Top, 53 lines --Top-----
Welcome to Info version 7.2. Type H for help, h for tutorial.
```

Назначение – просмотр и навигация по документам GNU (типа как man- но подробнее)

---

- Вызовите встроенную справку по программе GNU Info с помощью клавиши **h**. Ознакомьтесь с основами работы со справочной системой и зафиксируйте основные комбинации клавиш, используемые для навигации и поиска в справочной системе, а именно:

а. клавиши построчного и постраничного листания вверх и вниз:  
ctrl-n, ctrl-p  
построчно - space / ctrl-v || delete / PgUp, PgDown

б. клавиши навигации по страницам справочной системы:

n (next) / p (previus)

u(up по иерархии)

i (последний посещенный)

t( начало top)

d ( оглавление)

g --name (переход к имени)

с. клавиши навигации по пунктам меню справочной системы:

m --name /

выбрать по цифрам 1-9

или entr на строке

d. клавиши навигации по перекрестным ссылкам справочной системы:

По ссылкам или пункт на странице – tab / shift-tab

Если строка на нужном месте то enter

f --name of ref –

назад к узлу - L

---

e. клавишу выхода из справочной системы:

q

---

f. клавиши поиска информации на странице справочной системы:

вперед по тексту s / назад по тексту g + ввод строки

или по индексу I (виртуальный индекс) индексы в одну структуру

---

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:

# Лабораторная работа 3. Пользовательское окружение. Пользовательский профиль

## Упражнение 3.1. Переменные окружения. Переменные командного интерпретатора

- Посмотрите значение переменных окружения с помощью команды **env**. Зафиксируйте значения переменных окружения PATH, LANG, TERM:

```
monadayzek@MacBook-Air-Monad ~ % env | egrep '^PATH|LANG|TERM='
```

```
TERM=xterm-256color
PATH=/Users/monadayzek/.rvm/gems/ruby-3.2.2/bin:/Users/monadayzek/.rvm/gems/ruby-3.2.2@global/bin:/Users/monadayzek/.rvm/rubies/ruby-3.2.2/bin:/Library/Java/JavaVirtualMachines/temurin-21.jdk/Contents/Home/bin:/usr/local/bin:/usr/local/opt/postgresql@15/bin:/opt/homebrew/opt/postgresql@15/bin:/Users/monadayzek/.rbenv/shims:/usr/local/bin:/System/Cryptexes/App/usr/bin:/usr/bin:/bin:/usr/sbin:/sbin:/var/run/com.apple.security.cryptexd/codex.system/bootstrap/usr/local/bin:/var/run/com.apple.security.cryptexd/codex.system/bootstrap/usr/appleinternal/bin:/opt/X11/bin:/Library/Apple/usr/bin:/Library/TeX/texbin:/usr/local/share/dotnet:~/dotnet/tools:/opt/homebrew/bin:/Library/Frameworks/Mono.framework/Versions/Current/Commands:/Users/monadayzek/Library/Application Support/JetBrains/Toolbox/scripts:/Users/monadayzek/Library/Application Support/Coursier/bin:/Users/monadayzek/.rvm/bin
LANG=C.UTF-8
```

- 
- 
- 
- Установите в переменную окружения PATH пустое значение. : **MY\_PATH="\$PATH; export PATH=""**
  - Проследите за реакцией различных команд (**date**, **man**, ...). Зафиксируйте произошедшие изменения:

```
monadayzek@MacBook-Air-Monad ~ % date
zsh: command not found: date
```

При пустом PATH – поиск этих файлов по имени не будет работать, только если по абсолютному пути вызывать !

```
/usr/bin/man man || /bin/date
```

- 
- 
- 
- Восстановите старое значение переменной окружения PATH.  
**export PATH="\$MY\_PATH"**
  - Установите в переменную окружения LANG значение любого языка в формате **язык\_[СТРАНА.[кодировка]]**.  
**MY\_LANG="\$LANG"; export LANG="ru\_RU.UTF-8"**
  - Проследите за реакцией различных команд (**date**, **man**, ...). Зафиксируйте произошедшие изменения: Теперь на русском

```
monadayzek@MacBook-Air-Monad ~ % date
суббота, 27 сентября 2025 г. 12:31:24 (MSK)
```

7. Восстановите старое значение переменной окружения LANG.
8. Установите в переменную окружения TERM значение **vt100**.

`MY_ZS="$TERM"; export TERM="vt100"`

9. Проследите за реакцией различных команд (**mcedit**, **man**, ...). Зафиксируйте произошедшие изменения:

Так как на макос – не установлен mcedit, я не смог проверить. Но по идеи, man – должен упростить форматирование текста (меньше цветов и т.д)

---

---

---

Однако можно проверить свойства терминала его описание, до vt100 (xterm) и после vt100:

10. Восстановите старое значение переменной окружения TERM.

```
monadazek@MacBook-Air-Monad ~ % tput longname; tput cols; tput lines  
dec vt100 (w/advanced video)101  
34  
monadazek@MacBook-Air-Monad ~ % export TERM="$OLD_TERM"  
  
monadazek@MacBook-Air-Monad ~ % tput longname; tput cols; tput lines  
xterm with 256 colors101  
34
```

11. Посмотрите значение внутренних переменных командного интерпретатора **bash** с помощью команды **set**. Зафиксируйте значение переменной PS1:

```
monadazek@MacBook-Air-Monad ~ % set | grep '^PS1='  
PS1='%n@%m %1~ %# '
```

---

---

---

12. Установите во внутреннюю переменную PS1 любое значение.

`OLD_PS1="$PS1; export PS1='%n@%m:%~ %# '` – сделал как юзер@хост:путь

13. Проследите за реакцией командного интерпретатора. Зафиксируйте произошедшие изменения:

Было: `monadayzek@MacBook-Air-Monad ~ %`

Стало: `monadayzek@MacBook-Air-Monad:~ %`

В macOS можно еще использовать PROMPT вместо PS1

---

---

## Упражнение 3.2. Конфигурационные файлы пользователя окружения. Пользовательский профиль

- Создайте файл .plan в домашнем каталоге пользователя. Запишите в него любую информацию.

`printf '%s\n' 'Пожалуйста Поставьте автоматом 5!.' > ~/.plan`  
– это перезапись, можно и добавить `>> ~/.plan`

- При помощи команды `finger` посмотрите и зафиксируйте изменения в свойствах пользователя student:

Так как это на основной ОС – создал нового юзера студент `sudo sysadminctl -addUser student -fullName "studentAyzekr" -password "9420"`

```
monadayzek@MacBook-Air-Monad ~ % finger student
Login: student                                     Name: studentAyzekr
Directory: /Users/student                           Shell: /bin/zsh
Last login Sat Sep 27 12:22 (MSK) on ttys000
No Mail.
No Plan.
```

---

---

- Отредактируйте файл `.bashrc` или `.bash_profile` в домашнем каталоге пользователя. Примечание: если используется интерпретатор отличный от bash, то данные файлы не будут считаны, используйте `.profile` или специфичный для вашего интерпретатора файл конфигурации.

Запишите в файл команду присвоения в переменной окружения LANG значение любого языка в формате **язык\_[СТРАНА.[кодировка]]**:

На макосе- zsh, там кладут в окружения `~/.zprofile`  
`Echo 'export LANG=ru_RU.UTF-8' >> ~/.zprofile`

---

---

4. Завершите сеанс пользователя. Войдите в систему с использованием учетной записи student.

**su -l student**

5. Проследите за реакцией различных команд (**date**, **man**, ...). Зафиксируйте произошедшие изменения:

При переключении в student, `~/.zprofile` не изменился для student, можно сделать это в самом пользователе student либо так:

`sudo sh -c 'echo "export LANG=ru_RU.UTF-8" >> /etc/zprofile'` или  
`sudo launchctl config user global LANG ru_RU.UTF-8`

---

- 
6. Отредактируйте файл .bash\_profile в домашнем каталоге пользователя. Запишите в него команду установки в переменную PS1 собственного строкового значения для приглашения командного интерпретатора:

Так же, в `~/.zprofile`, `~/.zshrc` – не получится сделать через админа. Можно так:  
`sudo sh -c 'echo "PROMPT=\"%n@%m %1~ %# \\\"\" >> /etc/zshrc'`

---

---

7. Завершите сеанс пользователя. Войдите в систему с использованием учетной записи student.

`su -l student`

8. Проследите за реакцией командного интерпретатора. Зафиксируйте произошедшие изменения:

`student@MacBook-Air-Monad:~ %`

---

---

Лабораторная работа зачтена:  
Дата:  
Подпись преподавателя:



# Лабораторная работа 4. Файлы и файловая система UNIX

## Упражнение 4.1. Изучение структуры файловой системы

- При помощи следующих команд: **pwd**, **cd**, **ls** отработайте основные операции навигации в файловой системе:

а. Перейдите в корневой каталог.

cd / – вернет в корневой

б. Спускаясь по дереву каталогов, пройдитесь по основным каталогам и подкаталогам, и ознакомьтесь с их содержанием.

Можно так : **ls** – смотрим что в директории, далее проходим (можно одним):

cd bin && ls cd .. /etc && ls cd .. /usr && ls cd .. /var && ls

```
Applications      etc          private        Users
bin              home         sbin           usr
cores            Library       System          var
dev               opt          tmp            Volumes
[                dd           launchctl     pwd
bash              df           link           realpath
cat               echo          ln             rm
chmod             ed           ls             rmdir
cp                expr         hostname       sh
csh               hostname     kill           sleep
dash              kill         mv             stty
date              ksh          pax            sync
afpovertcp.cfg
aliases
aliases.db
apache2
```

- с. Зафиксируйте структуру иерархии файловой системы и назначение основных каталогов:

```
monadayzek@MacBook-Air-Monad ~ % tree -L 1/
.
├── Applications
├── bin
├── cores
├── dev
├── etc    -> private/etc
├── home   -> /System/Volumes/Data/home
├── Library
├── opt
└── private
    ├── sbin
    └── System
        ├── tmp    -> private/tmp
        └── Users
            ├── usr
            └── var    -> private/var
└── Volumes

17 directories, 0 files
```

Можно в макос поставить через brew – tree и с корневой директории вызвать ее увидев вообще всю иерархию  
Но так как стоит пожалеть ноутбук, в tree можно указывать уровни с аргументом L : tree -L 1/

## Упражнение 4.2. Типы файлов

1. Для экземпляров различных типов файлов:
  - a. обычный файл (/etc/fstab, /bin/bash, /usr/bin/apropos)
  - b. каталог (/tmp)
  - c. специальный файл устройства (/dev/sda, /dev/tty1)
  - d. именованный канал (/run/systemd/initctl fifo)
  - e. сокет (/run/systemd/journal/dev-log)
  - f. символьическая ссылка (/dev/cdrom)

Примечание: при отсутствии файлов из перечня найдите аналог соответствующего типа.

2. Запишите особенности отображения различных типов файлов командой **ls -l**:

```
monadazek@MacBook-Air-Monad / % ls -l /bin/zsh /bin/ls /usr/bin/man  
-rwxr-xr-x 1 root wheel 154624 9 сент. 10:15 /bin/ls  
-rwxr-xr-x 1 root wheel 1361200 9 сент. 10:15 /bin/zsh  
-r-xr-xr-x 4 root wheel 32136 9 сент. 10:15 /usr/bin/man
```

Каталог:

```
monadazek@MacBook-Air-Monad / % ls -ld /tmp  
lrwxr-xr-x@ 1 root wheel 11 9 сент. 10:15 /tmp -> private/tmp
```

Терминал:

```
crw--w---- 1 monadazek tty 0x10000000 28 сент. 17:15 /dev/ttys000
```

Диск:

```
brw-r----- 1 root operator 0x10000000 24 сент. 14:10 /dev/disk0
```

Именнованый канал FIFO

```
monadazek@MacBook-Air-Monad / % ls -l ~/myfifo  
prw-r--r-- 1 monadazek staff 0 28 сент. 17:16 /Users/monadazek/myfifo
```

Сокет:

```
srw-rw-rw- 1 root daemon 0 24 сент. 14:10 /var/run/syslog
```

Символ. Ссылка :

```
brw-r----- 1 root operator 0x10000000 24 сент. 14:10 dev/disk0
```

3. Используя команды: **stat**, **file** исследуйте метаданные и содержимое различных типов файлов (из [Шаг 1](#)):

```
monadazek@MacBook-Air-Monad / % stat /bin/zsh  
file /bin/zsh  
stat ~/myfifo  
file ~/myfifo  
  
16777230 1152921500312554172 -rwxr-xr-x 1 root wheel 0 1361200 "Sep 9 10:15:49 2025" "Sep 9 10:15:49 2025" "Sep 9 10:15:49 2025" "Sep 9 10:15:49 2025" 4096 1472 0x80020 /bin/zsh  
/bin/zsh: Mach-O universal binary with 2 architectures: [x86_64:Mach-O 64-bit executable x86_64] [arm64e:Mach-O 64-bit executable arm64e]  
/bin/zsh (for architecture x86_64): Mach-O 64-bit executable x86_64  
/bin/zsh (for architecture arm64e): Mach-O 64-bit executable arm64e  
16777230 10787578 prw-r--r-- 1 monadazek staff 0 0 "Sep 28 17:16:56 2025" "Sep 28 17:16:56 2025" "Sep 28 17:16:56 2025" "Sep 28 17:16:56 2025" 4096 0 0 /Users/monadazek/myfifo  
/Users/monadazek/myfifo: fifo (named pipe)
```



## Упражнение 4.3. Манипулирование объектами дерева каталогов UNIX

- При помощи следующих команд: **touch**, **mkdir**, **rmdir**, **cp**, **mv**, **rm**, **ln**, **mknod** отработайте основные операции работы с файловой системой, зафиксируйте полученные результаты:

a. перейдите в домашний каталог пользователя: **на маках можно как cd ~**

```
monadayzek@MacBook-Air-Monad ~ % pwd  
/Users/monadayzek
```

b. получите информацию о текущем рабочем каталоге:

ls -la :

```
monadayzek@MacBook-Air-Monad ~ % ls -la  
total 464  
drwxr-x---+ 85 monadayzek staff 2720 28 сент. 17:16 .  
drwxr-xr-x 6 root admin 192 27 сент. 12:53 ..  
drwxr-xr-x@ 17 monadayzek staff 544 17 сент. 15:57 .anydesk  
drwxr-xr-x@ 4 monadayzek staff 128 27 авг. 23:24 .aspnet  
-rw----- 1 monadayzek staff 555 22 сент. 12:38 .bash_history  
-rw-r--r-- 1 monadayzek staff 420 8 авг. 01:36 .bash_profile  
-rw-r--r-- 1 monadayzek staff 118 8 авг. 01:36 .bashrc  
drwxr-xr-x@ 3 monadayzek staff 96 31 июля 22:08 .bundle  
drwxr-xr-x 3 monadayzek staff 96 28 сент. 00:26 .cache  
-r----- 1 monadayzek staff 7 16 июля 23:58 .CFUserTextEncoding  
drwxr-xr-x 6 monadayzek staff 192 4 сент. 12:56 .colima  
drwxr-xr-x@ 9 monadayzek staff 288 13 сент. 18:47 .config  
drwxr-xr-x@ 5 monadayzek staff 160 31 авг. 02:40 .cursor  
drwxr-xr-x 7 monadayzek staff 224 4 сент. 12:56 .docker  
drwxr-xr-x@ 20 monadayzek staff 640 29 авг. 11:59 .dotnet
```

c. получите информацию о содержимом текущего каталога: ВЫШЕ ^

d. создайте новый подкаталог subdir в текущем каталоге:

**mkdir subdir**

e. перейдите в него: **cd subdir**

f. создайте в данном подкаталоге новый файл myfirstfile:

vim myfirstfile / touch myfirstfile / emacs myfirstfile

g. создайте мягкую symlink и жесткую hardlink ссылки на него:

ln -s myfirstfile myfirstfile\_symlink

ln myfirstfile myfirstfile\_hardlink

h. ~~переместите и/или скопируйте файл myfirstfile в вышележащий каталог:~~

~~скопирует - cp myfirstfile ..~~

~~переместит - mv myfirstfile ..~~

- 
- i. удалите оставшиеся файлы:  
`rm myfirstfile_symlink myfirstfile_hardlink`
- 

- j. перейдите в вышележащий каталог и удалите созданный вами подкаталог subdir и файл myfirstfile:  
`cd ..` – переводит по иерархии выше  
`rm -r subdir`  
`rm myfirstfile`
- 

- k. создайте именованный канал myfifo: `mkfifo myfifo`
- 
- 

- l. создайте образ оптического диска, находящийся в накопителе /dev/cdrom:

На макос нет /cdrom, но есть аналог, если есть устройство типа флешки, то она будет в /Volumes/O\_O (моя флешка)  
`hdiutil create -srcdevice /dev/disk2 ~/cdrom.dmg` – создастся образ диска .dmg

---

## Упражнение 4.4. Поиск файлов

- При помощи следующих команд: **find**, **whereis**, **which** отработайте основные операции поиска файлов, зафиксируйте полученные результаты:
  - найдите в каталоге /usr все файлы размером больше 800Kb

Так как нереально вывести скрин, можно командой tee записать все в .txt файл :  
`find /usr -type f -size +800k 2>/dev/null | tee osunix.txt`

- b. найдите информацию о всех исполняемых файлах, страницах руководства и исходных текстах программы **ls**

Сначала что б понять где лежит ls – whereis ls  
Потом man ls

- c. найдите информацию о местоположении программ **bash**, **passwd**, **chsh**, **chfn**, **finger**

```
monadazek@MacBook-Air-Monad ~ % which bash passwd chsh finger
whereis bash passwd chsh finger

/bin/bash
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/finger
bash: /bin/bash /usr/share/man/man1/bash.1
passwd: /usr/bin/passwd /usr/share/man/man1/passwd.1
chsh: /usr/bin/chsh /usr/share/man/man1/chpass.1
finger: /usr/bin/finger /usr/share/man/man1/finger.1
```

Можно как с which можно и с whereis

---

---

## Упражнение 4.5. Операции с файловыми системами

1. При помощи следующих команд: **mount**, **umount**. Примечание: может потребоваться повышение привилегий до суперпользователя:

- a. примонтируйте файловую систему находящуюся на оптическом диске или флеш-накопителе, в каталог /mnt, проследите за сообщениями:

```
mkdir: /mnt: Read-only file system
/dev/disk0 (internal, physical):
#:          TYPE NAME                      SIZE IDENTIFIER
0: GUID_partition_scheme                  *251.0 GB disk0
1:      Apple_APFS_ISC Container disk1    524.3 MB disk0s1
2:      Apple_APFS Container disk2        245.1 GB disk0s2
3:      Apple_APFS_Recovery Container disk3   5.4 GB disk0s3

/dev/disk2 (synthesized):
#:          TYPE NAME                      SIZE IDENTIFIER
0: APFS Container Scheme -                +245.1 GB disk2
                                         Physical Store disk0s2
1:      APFS Volume Macintosh HD           12.0 GB disk2s1
2:      APFS Snapshot com.apple.os.update-... 12.0 GB disk2s1s1
3:      APFS Volume Preboot              7.8 GB disk2s2
4:      APFS Volume Recovery             1.2 GB disk2s3
5:      APFS Volume Macintosh HD - Data  144.0 GB disk2s5
6:      APFS Volume VM                  2.1 GB disk2s6

/dev/disk4 (external, physical):
#:          TYPE NAME                      SIZE IDENTIFIER
0: FDisk_partition_scheme                *63.3 GB disk4
1:      DOS_FAT_32 0_0                     63.3 GB disk4s1
```

```
sudo mkdir -p /mnt
diskutil list
diskutil mount -t apfs /dev/disk4s1#
```

- b. размонтируйте находящуюся на оптическом диске или флеш-накопителе файловую систему, примонтированную в каталог /mnt, и проследите за сообщениями: **sudo diskutil unmount /mnt**

2. При помощи следующих команд: **du**, **df** подсчитайте:

- a. место, занимаемое на диске домашнего каталога пользователя:

```
du -sh ~ : 49G /Users/monauvezek
```

- b. место, занимаемое на диске каталогом /tmp – **du -sh /tmp** (выдал 0байт )

- c. размер корневой файловой системы

Так как я делаю лабораторные на основной ОС- это займет слишком много времени и лучше посмотреть через GUI, идея : **df -h /** будет примерно 165 гб

- d. размер файловой системы оптического диска или флеш-накопителя

```
df -h /disk4s1#
```

```
monadayzek@MacBook-Air-Monad ~ % df -h /dev/disk4s1#
Filesystem      Size   Used  Avail Capacity iused ifree %iused  Mounted on
/dev/disk4s1    59Gi   5,0Mi  59Gi     1%        0      0       -  /Volumes/0_0
monadayzek@MacBook-Air-Monad ~ %
```

---

e. свободное место, оставшееся на корневой файловой системе (AVAIL)

```
monadayzek@MacBook-Air-Monad ~ % df -h /
Filesystem      Size   Used  Avail Capacity iused ifree %iused  Mounted on
/dev/disk2s1s1   228Gi  11Gi   72Gi    14%     447k  759M    0%   /
monadayzek@MacBook-Air-Monad ~ %
```

---

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:

# Лабораторная работа 5. Владельцы и права доступа объектов файловой системы UNIX.

## Упражнение 5.1. Владельцы файлов

- Зафиксируйте права доступа к файлу и владельцев файлов из [Шаг 1 упр. 4.2, лаб. 4:](#)

```
monadayzek@MacBook-Air-Monad ~ % ls -l /bin/ls /bin/bash /usr/bin/man
-r-xr-xr-x 1 root wheel 1310208 25 сент. 10:03 /bin/bash
-rwxr-xr-x 1 root wheel 154624 25 сент. 10:03 /bin/ls
-r-xr-xr-x 4 root wheel 32136 25 сент. 10:03 /usr/bin/man
monadayzek@MacBook-Air-Monad ~ % ls -l /tmp
lrwxr-xr-x@ 1 root wheel 11 25 сент. 10:03 /tmp -> private/tmp
monadayzek@MacBook-Air-Monad ~ % ls -l /dev/disk0 /dev/ttys000
brw-r---- 1 root operator 0x1000000 29 сент. 23:16 /dev/disk0
crw--w--- 1 monadayzek tty 0x10000000 4 окт. 15:41 /dev/ttys000
monadayzek@MacBook-Air-Monad ~ % ls -l /var/run/utmpx
-rw-r--r--@ 1 root daemon 5024 4 окт. 15:34 /var/run/utmpx
monadayzek@MacBook-Air-Monad ~ % ls -l /var/run/syslog
srw-rw-rw- 1 root daemon 0 29 сент. 23:18 /var/run/syslog
monadayzek@MacBook-Air-Monad ~ % ls -l /dev/console
crw--w--w- 1 monadayzek staff 0 1 окт. 22:32 /dev/console
monadayzek@MacBook-Air-Monad ~ %
```

- 
- 
- 
- При помощи команд: **chown**, **chgrp** попробуйте подарить файлы из домашнего каталога пользователя *student* пользователю *vinnie*, группе *root*. Зафиксируйте реакцию системы:

Так как делаю на родной ОС, пришлось заново создать винни

```
monadayzek@MacBook-Air-Monad ~ % sudo sysadminctl -addUser Vinnie 9420
Password:
2025-10-04 15:42:34.703 sysadminctl[34569:2308961] -----
2025-10-04 15:42:34.704 sysadminctl[34569:2308961] No clear text password or interactive option was specified (adduser, change/reset password will not allow user to use FDE) !
2025-10-04 15:42:34.704 sysadminctl[34569:2308961] -----
2025-10-04 15:42:34.878 sysadminctl[34569:2308961] Creating user record...
2025-10-04 15:42:34.910 sysadminctl[34569:2308961] Assigning UID: 503 GID: 20
2025-10-04 15:42:37.481 sysadminctl[34569:2308961] Creating home directory at /Users/Vinnie
```

Сначала нужно узнать homedir у студента – это *Users/student*, потом в *HOMEDIR* нужно поместить этот каталог : *HOMEDIR=\$(dscl . -read /Users/student NFSHomeDirectory 2>/dev/null | awk '{print \$2}')*

Затем уже даем *vinnie* Homedir

*sudo chown -vR vinnie "\$HOMEDIR"* – рекурсивно меняет владельца и показывает что меняется

У macos – нет группы *root*, есть аналог – *wheel*. Для него так же можно заменив *vinnie* на *wheel*:

*sudo chgrp -vR wheel "\$HOMEDIR"*

## Упражнение 5.2. Права доступа к объектам файловой системы UNIX

- Создайте новый файл newfile. Зафиксируйте права доступа к файлу и владельцев полученного файла:

```
monadazek@MacBook-Air-Monad ~ % touch newfile  
ls -l newfile  
  
-rw-r--r-- 1 monadazek staff 0 4 окт. 15:55 newfile
```

Mkdir test  
755

---

---

---

- При помощи команды: **umask**:

- просмотрите и зафиксируйте значение реверсной маски прав доступа по умолчанию для вновь создаваемых файлов

У макос – по умолчанию стоит всегда маска 022, то есть биты которые будут убраны при создании новых -f, -d -> -rw-r--r-- | drwxr-x-r-x

- установите значение реверсной маски прав доступа по умолчанию в 000. Создайте файл defaultfile. Зафиксируйте права доступа к файлу и владельцев полученного файла:

Для маски 000 – разрешает все возможные варианты доступа, мы создали файл так что:

```
monadazek@MacBook-Air-Monad ~ % umask 000  
touch defaultfile  
ls -l defaultfile  
  
-rw-rw-rw- 1 monadazek staff 0 4 окт. 15:58 defaultfile
```

---

---

- с. установите значение реверсной маски прав доступа по умолчанию в 777. Создайте файл maskedfile. Зафиксируйте права доступа к файлу и владельцев полученного файла:

```
monadazek@MacBook-Air-Monad ~ % umask 777
touch maskedfile
ls -l maskedfile

----- 1 monadazek staff 0 4 окт. 16:00 maskedfile
monadazek@MacBook-Air-Monad ~ %
Вышел полный запрет ._.
```

- 
- 
3. При помощи команды: **chmod** отработайте основные операции изменения владельцев и прав доступа к файлам:

- a. создайте в домашнем каталоге файл privatefile и установите права доступа rw- ---:

```
monadazek@MacBook-Air-Monad ~ % touch ~/privatefile
chmod 600 ~/privatefile
ls -l ~/privatefile

-rw----- 1 monadazek staff 0 4 окт. 16:02 /Users/monadazek/privatefile
600 - дает права владельцу на rw
```

- b. создайте в домашнем каталоге файл publicfile и установите права доступа rw- r--:

```
monadazek@MacBook-Air-Monad ~ % touch ~/publicfile
chmod 644 ~/publicfile
ls -l ~/publicfile

-rw-r--r-- 1 monadazek staff 0 4 окт. 16:02 /Users/monadazek/publicfile
644- дает владельцу - читать писать а другим просто читать
```

- c. создайте в домашнем каталоге новый каталог groupsubdir и установите права доступа rw- r-- ---:

```
monadazek@MacBook-Air-Monad ~ % mkdir ~/groupsubdir
chmod 640 ~/groupsubdir
ls -ld ~/groupsubdir

drw-r---- 2 monadazek staff 64 4 окт. 16:03 /Users/monadazek/groupsubdir
А тут уже создаем директорию и 640 дает права без X, то есть нельзя зайти и прочесть содержимое
```

- d. создайте в домашнем каталоге новый каталог publicdir и установите права доступа rw- rw- rw-:

```
monadayzek@MacBook-Air-Monad ~ % mkdir ~/publicdir  
chmod 666 ~/publicdir  
ls -ld ~/publicdir  
  
drw-rw-rw- 2 monadayzek staff 64 4 окт. 16:04 /Users/monadayzek/publicdir
```

666 – дает права так: директория не доступна но чтение и запись для всех. То есть если по нормальному то 777

---

---

Лабораторная работа зачтена:  
Дата:  
Подпись преподавателя:

# Лабораторная работа 6. Обработка текстовой информации. Управление вводом-выводом команд. Подстановки командного интерпретатора

## Упражнение 6.1. Обработка текстовой информации в UNIX

- При помощи следующих команд: **cat**, **less**, **more**, **tail**, **head**, **od**, **hexdump**, **cmp**, **comm**, **diff**, **cut**, **grep**, **sort**, **wc**, **tr** отработайте основные операции работы с содержимым файлов, зафиксируйте полученные результаты:

а. выведите текстовый файл (`/etc/protocols`) на экран полностью и по частям:

```
monadayzek@MacBook-Air-Monad ~ % cat /private/etc/protocols
#
# Internet protocols
#
# $FreeBSD$
#      from: @(#)protocols      5.1 (Berkeley) 4/17/89
#
# See also http://www.iana.org/assignments/protocol-numbers
#
ip      0      IP          # internet protocol, pseudo protocol number
#hopopt 0      HOPOPT       # hop-by-hop options for ipv6
icmp    1      ICMP         # internet control message protocol
igmp    2      IGMP         # internet group management protocol
ggp     3      GGP          # gateway-gateway protocol
ipencap 4      IP-ENCAP    # IP encapsulated in IP (officially ``IP'')
st2     5      ST2          # ST2 datagram mode (RFC 1819) (officially ``ST'')
')
tcp     6      TCP          # transmission control protocol
cbt     7      CBT          # CBT, Tony Ballardie <A.Ballardie@cs.ucl.ac.uk>
egp     8      EGP          # exterior gateway protocol
igp     9      IGP          # any private interior gateway (Cisco: for IGRP)
bbn-rcc 10     BBN-RCC-MON # BBN RCC Monitoring
nvp    11     NVP-II       # Network Voice Protocol
pup    12     PUP          # PARC universal packet protocol
argus   13     ARGUS        # ARGUS
emcon   14     EMCON        # EMCON
xnet    15     XNET         # Cross Net Debugger
```

Cat – выведет весь файл, less/more – по страницам, head по дефолту ласт 10, tail конец строк 10

---

---

---

б. выведите двоичный файл (`/bin/bash`) в шестнадцатеричном и восьмеричном виде:

```
monadayzek@MacBook-Air-Monad ~ % hexdump -C /bin/zsh | head
00000000  ca fe ba be 00 00 00 02 01 00 00 07 00 00 00 03
00000010  00 00 40 00 00 09 c3 e0 00 00 00 0e 01 00 00 0c
00000020  80 00 00 02 00 0a 40 00 00 0a 85 30 00 00 00 0e
00000030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
00004000  cf fa ed fe 07 00 00 01 03 00 00 00 02 00 00 00
00004010  15 00 00 00 f0 06 00 00 85 00 20 00 00 00 00 00
00004020  19 00 00 00 48 00 00 00 5f 5f 50 41 47 45 5a 45
00004030  52 4f 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004040  00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00
monadayzek@MacBook-Air-Monad ~ % od -x /bin/zsh | head
00000000  fec a  be b a 0000 0200 0001 0700 0000 0300
00000020  0000 0040 0900 e0c3 0000 0000 0001 0001 0000
00000040  0080 0200 0a00 0040 0a00 3085 0000 0000 0000
00000060  0000 0000 0000 0000 0000 0000 0000 0000 0000
*
00400000  fac f feed 0007 0100 0003 0000 0002 0000
0040020  0015 0000 06f0 0000 0085 0020 0000 0000 0000
0040040  0019 0000 0048 0000 5f5f 4150 4547 455a
0040060  4f52 0000 0000 0000 0000 0000 0000 0000 0000
0040100  0000 0000 0001 0000 0000 0000 0000 0000 0000
monadayzek@MacBook-Air-Monad ~ % od -o /bin/zsh | head
00000000  177312 137272 000000 001000 000001 003400 000000 001400
00000020  000000 000100 004400 160303 000000 007000 000001 006000
00000040  000200 001000 005000 000100 005000 030205 000000 007000
00000060  000000 000000 000000 000000 000000 000000 000000 000000
*
00400000  175317 177355 000007 000400 000003 000000 000002 000000
0040020  000025 000000 003360 000000 000205 000040 000000 000000
0040040  000031 000000 000110 000000 057537 040520 042507 042532
0040060  047522 000000 000000 000000 000000 000000 000000 000000
0040100  000000 000000 000001 000000 000000 000000 000000 000000
```

На макос – есть hexdump (дает hex и ASCII), od -o (дает 16-но), od-x (дает 8-но) (вывожу ZSH)

---

---

---

с. сравните текстовые (/etc/hosts.allow и /etc/hosts.deny) и двоичные (/bin/bash и /bin/sh) файлы:

```
monadazek@MacBook-Air-Monad ~ % diff /private/etc/services /private/etc/protocols
ls
cmp /private/etc/services /private/etc/protocols
comm -12 <(sort /private/etc/services) <(sort /private/etc/protocols)

2.13926c2.158
< # Network services, Internet style
< #
< # Note that it is presently the policy of IANA to assign a single well-known
< # port number for both TCP and UDP; hence, most entries here have two entries
< # even if the protocol doesn't support UDP operations.
< #
< # The latest IANA port assignments can be gotten from
< #
< # http://www.iana.org/assignments/port-numbers
< #
< # The Well Known Ports are those from 0 through 1023.
< # The Registered Ports are those from 1024 through 49151
< # The Dynamic and/or Private Ports are those from 49152 through 65535
< #
< # $FreeBSD: src/etc/services,v 1.89 2002/12/17 23:59:10 eric Exp $
< #     From: @(#)services      5.8 (Berkeley) 5/9/91
< #
< # WELL KNOWN PORT NUMBERS
< #
< rtmp          1/ddp    #Routing Table Maintenance Protocol
< tcpxm         1/udp    # TCP Port Service Multiplexer
< tcpxm         1/tcp    # TCP Port Service Multiplexer
< #                         Mark Lottor <MKL@nisc.sri.com>
< nbp          2/ddp    #Name Binding Protocol

```

/private/etc/services /private/etc/protocols differ: char 5, line 2

```
#
```

```
#
```

БИН:

```
#
```

```
monadazek@MacBook-Air-Monad ~ % diff /bin/zsh /bin/sh
cmp /bin/zsh /bin/sh
```

```
Binary files /bin/zsh and /bin/sh differ
/bin/zsh /bin/sh differ: char 22, line 1
```

На макос – нет hosts.allow/deny – но можно сравнивать так : 1) построчное различие (diff), cmp побайтное отличие, дает первые отличные байты / строку) 3) comm (пересечение отсортированных файлов (12 дает только общие):

d. найдите все строки файла (/etc/services), содержащие слово **file**, затем строки содержащие **четырехзначные** числа, затем строки не начинающиеся со знака **#**:

```
monadazek@MacBook-Air-Monad ~ % grep 'file' /private/etc/services
grep -v '^#' /private/etc/services
      59/udp  # any private file service
      59/tcp   # any private file service
profile  136/udp  # PROFILE Naming System
profile  136/tcp   # PROFILE Naming System
efs      520/tcp  # extended file name server
gpfss   564/udp  # pfns file service
gpfss   564/tcp   # plan 9 file service
#
rfile    750/tcp  # Clay Maeckel <clay_maeckel@filemaker.com>
bnetfile 1120/tcp  # Battle.net File Transfer Protocol
bnetfile 1120/udp # Battle.net File Transfer Protocol
filex-lport 1887/udp # FileX Listening Port
filex-lport 1887/tcp # FileX Listening Port
simple-tx-rx 2257/tcp # simple text/file transfer
simple-tx-rx 2257/udp # simple text/file transfer
mfserver  2266/udp # M-Files Server
#
# Clay Maeckel <clay_maeckel@filemaker.com>
Yusuke Asai <yasai@nw1.file.fc.nec.co.jp>
mobile-file-dl 2926/udp # MOBILE-FILE-DL
mobile-file-dl 2926/tcp # MOBILE-FILE-DL
filecast  3401/udp # filecast
filecast  3401/tcp # filecast
csrc     3417/udp # ConservR file translation
csrc     3417/tcp # ConservR file translation
rfa      4672/udp # remote file access server
rfa      4672/tcp # remote file access server
profilemac 4749/tcp # Profile for Mac
profilemac 4749/udp # Profile for Mac
#
Clay Maeckel <clay_maeckel@filemaker.com>
avt-profile-1 5004/udp # avt-profile-1
avt-profile-1 5004/tcp # avt-profile-1
avt-profile-2 5005/udp # avt-profile-2
avt-profile-2 5005/tcp # avt-profile-2
afs3-fileserver 7000/udp # afs3 file server itself
afs3-fileserver 7000/tcp # file server itself
mindfilesys  7391/udp # mind-file system server
mindfilesys  7391/tcp # mind-file system server
isdd     8148/tcp # i-SDD file transfer
```

Тут # - коммент

Grep -E '\b[0-9]{4}\b'

е. подсчитайте количество строк/слов/букв в файле (/etc/protocols):

```
monadayzek@MacBook-Air-Monad ~ % wc /private/etc/protocols
```

```
158    1071   6393 /private/etc/protocols
```

```
wc monadayzek@MacBook-Air-Monad ~ %
```

f. отсортируйте содержимое файла (/etc/rpc):

```
#  
#  
# $FreeBSD$  
# rpc 88/08/01 4.0 RPCSRC: from 1.12 99/07/25 SMI  
# The range 20010000-20019999 is reserved for Apple services  
3270_mapper 100013  
activity 100109 na.activity # SunNet Manager  
alis 100018  
amd 300019  
bootparamd 100026 bootparam  
bugtraqd 100071  
cmsgd 100068  
database_svc 100016  
debug_svc 100066 dbsrv  
etherif 100118 na.etherif  
etherstatd 100010 etherstat  
event 100101 na.event # SunNet Manager  
hostif 100117 na.hostif  
hostmem 100112 na.hostmem  
hostperf 100107 na.hostperf  
ioadmd 100055 rpc.ioadmd  
iproutes 100120 na.iproutes  
kerbd 100078  
keyserv 100029 keyserver  
layers 100121 na.layers  
llockmgr 100020  
logger 100102 na.logger # SunNet Manager  
mountd 100005 mount showmount  
netinfo 200100000  
netinfobind 200100001  
NETlicense 100062  
nfs 100003 nfsprog  
nfs_acl 100227  
nisd 100300 rpc.nisd  
nispasswd 100303 rpc.nispasswd  
nlockmgr 100021  
nsed 100038  
nsemntd 100039  
pcnfsd 150001 pcnfs  
ping 100115 na.ping  
rxd 100017 rex  
rje_mapper 100014  
rpcbind 100000 portmap sunrpc rpcbind  
rpcnfs 100116 na.rpcnfs  
rquotad 100011 rquotaproq quota rquota  
rstatd 100001 rstat rstat_svc rup perfmeter  
rusersd 100002 rusers  
sadmind 100232
```

sort /private/etc/rpc | less

---

---

---

g. извлеките из структурированного файла (/etc/passwd) первое поле всех строк:

```
monadayzek@MacBook-Air-Monad ~ % cut -d: -f1 /private/etc/passwd  
##  
# User Database  
#  
# Note that this file is consulted directly only when the system is running  
# in single-user mode. At other times this information is provided by  
# Open Directory.  
#  
# See the opendirectoryd(8) man page for additional information about  
# Open Directory.  
##  
nobody  
root  
daemon  
_uucp  
_taskgated  
_networkd  
_installassistant  
_lp  
_postfix  
_scsd  
_ces  
_appstore  
_mcxalr  
_appleevents  
_geod  
_devdocs  
_sandbox
```

: - разделитель, тут взяли 1-ое поле – имена пользователей

---

---

## Упражнение 6.2. Перенаправление ввода-вывода информации

- При помощи конструкций перенаправления: >, <, >>, | и соответствующих команд, отработайте основные операции конвейерной обработки информации:

- a. вырежьте из файла **/etc/passwd** пятое поле, отсортируйте в обратном порядке и сохраните в файл **users** : **cut -d: -f5 /private/etc/passwd | sort -r >> users**

```
Asset Cache Service
Application Server
Application Owner
applepay Account
AppleNeuralEngine
AppleEvents Daemon
Apple Virtual Platform HID Bridge
Apple Remote Desktop
Apple Events User
App Install Daemon
Analytics Daemon
AMaViS Daemon
Always On Sense Daemon
Accessory Update Daemon
##
##
# User Database
# See the opendirectoryd(8) man page for additional information about
# Open Directory.
# Open Directory.
# Note that this file is consulted directly only when the system is running
# in single-user mode. At other times this information is provided by
#
#
_LAUNCHSERVICESD
```

- b. преобразуйте содержимое файла **users** к верхнему регистру  
**tr [:lower:] [:upper:] < users >> users\_upper**

```
APPLENEURALENGINE
APPLEEVENTS DAEMON
APPLE VIRTUAL PLATFORM HID BRIDGE
APPLE REMOTE DESKTOP
APPLE EVENTS USER
APP INSTALL DAEMON
ANALYTICS DAEMON
AMAVIS DAEMON
ALWAYS ON SENSE DAEMON
ACCESSORY UPDATE DAEMON
##
##
# USER DATABASE
# SEE THE OPENDIRECTORYD(8) MAN PAGE FOR ADDITIONAL INFORMATION ABOUT
# OPEN DIRECTORY.
# OPEN DIRECTORY.
# NOTE THAT THIS FILE IS CONSULTED DIRECTLY ONLY WHEN THE SYSTEM IS RUNNING
# IN SINGLE-USER MODE. AT OTHER TIMES THIS INFORMATION IS PROVIDED BY
#
#
_LAUNCHSERVICESD
```

- c. найдите все регулярные файлы в принадлежащие пользователю **student** и сохраните найденный список в файле **myfiles**, ошибки поиска сохраните в файле **errors**:

```
find: /Users/monadayzek/Library/Suggestions: Operation not permitted
find: /Users/monadayzek/Library/Weather: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.stocks-news: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.photolibraryd.private: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.FullKeyboardAccess: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.feedback: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.siri.inference: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.telephonyutilites.callservicesd: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.swtransparency: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.coreservices.usreactivityd: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.ArchiveUtility.PKSignedContainer: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.accessibility.voicebanking: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.icloud.searchpartyuseragent: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.siri.referenceResolution: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.stocks: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.usernoted: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.VoiceMemos.shared: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.contacts: Operation not permitted
find: /Users/monadayzek/Library/Group Containers/group.com.apple.secure-control-center-preferences: Operation not permitted
```

monadayzek@MacBook-Air-Monad ~ % sudo find / -xdev -type f -user student > myfiles 2>errors

- Очень долго так как это основная ОС
- 
- 

- d. найдите все каталоги принадлежащие пользователю **student** и добавьте полученный список в файл **myfiles**, подавите вывод сообщений об ошибках поиска

monadayzek@MacBook-Air-Monad ~ % sudo find / -xdev -type d -user student >> myfiles

2>/dev/null

```
monadayzek@MacBook-Air-Monad ~ % cat myfiles
/System/Volumes/Data/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/mdsDirectory.db
/System/Volumes/Data/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/lock
/System/Volumes/Data/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/mdsObject.db
/System/Volumes/Data/Users/student/Library/Containers/com.apple.ImageIO.imageImporter/.com.apple.containermanagerd.metadata.plist
/System/Volumes/Data/Users/student/Library/Containers/com.apple.PDFKit.PDFImporter/.com.apple.containermanagerd.metadata.plist
/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/mdsDirectory.db
/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/lock
/private/var/folders/fk/46pl8pc537l7g07ztcxvhlswo0000gp/C/com.apple.trustd/mds/mdsObject.db
/Users/student/Library/Containers/com.apple.ImageIO.imageImporter/.com.apple.containermanagerd.metadata.plist
/Users/student/Library/Containers/com.apple.PDFKit.PDFImporter/.com.apple.containermanagerd.metadata.plist
/Library/Caches/Desktop Pictures/74886B4D-1661-4679-9DE0-3FF4CEEE13CF
/System/Volumes/Data/Library/Caches/Desktop Pictures/74886B4D-1661-4679-9DE0-3FF4CEEE13CF
```

е. подсчитайте суммарное количество обычных файлов в дереве каталогов UNIX

```
monadayzek@MacBook-Air-Monad ~ % sudo find / -xdev -type f 2>/dev/null | wc -l
```

```
3085712
```

## Упражнение 6.3. Подстановка вывода команд

- При помощи конструкций подстановки вывода команд `` , \$( ) отработайте основные операции подстановки вывода команд:

а. Сохраните в переменной DATE текущую дату и время

```
monadayzek@MacBook-Air-Monad ~ % DATE="$(date '+%Y-%m-%d %H:%M:%S %z')"
```

```
monadayzek@MacBook-Air-Monad ~ % echo $DATE  
2025-10-04 16:59:05 +0300
```

- б. Получите информацию о суммарном количестве строчек во всех регулярных файлах домашнего каталога пользователя:

TOTAL\_LINES=\$(find ~ -type f -exec wc -l {} + 2>/dev/null | awk {s+=\$1})" – но это будет очень долго так как эта основная ОС. Идея – find – возьмет все файлы, wc -l посчитает строки, awk – просуммирует, ошибки доступа уйдут в dev/null // без тотал

- с. создайте файл, в имени которого присутствует текущая дата и время.

```
monadayzek@MacBook-Air-Monad ~ % touch "$HOME/data_sev$(date +%Y%m%d_%H%M%S)"
```

```
monadayzek@MacBook-Air-Monad ~ % ls  
Applications           Music  
data_sev20251004_170646.txt myfifo  
defaultfile            myfiles  
Desktop                newfile  
Documents              Pictures  
Downloads              privatefile  
errors                 Public  
forVinnie               publicdir  
groupsubdir             publicfile  
Library                Quantum_computing.ipynb  
maskedfile              users  
Movies                 users_upper
```

## **Упражнение 6.4. Подстановка шаблонных символов**

- При помощи шаблонных символов: \*, ? и соответствующих команд:
    - выведите полную информацию о подкаталогах домашнего каталога пользователя, заканчивающихся символами **dir** и о файлах содержащих в своем названии подстроку **file**:

Взял директории, . – обычные файлы, N – даже если есть совпадения – пропустит, ls -ld – права владелец группа

```
monadayzek@MacBook-Air-Monad ~ % ls -ld ~/dir(/N) ~/file*(.N)
```

```
-rw-rw-rw- 1 monadayzek staff 0 4 okt. 15:58 /Users/monadayzek/defaultfile  
drw-r---- 2 monadayzek staff 64 4 okt. 16:03 /Users/monadayzek/groupsubdir  
----- 1 monadayzek staff 0 4 okt. 16:00 /Users/monadayzek/maskedfile  
-rw-r--r-- 1 monadayzek staff 3756 4 okt. 16:47 /Users/monadayzek/myfiles  
-rw-r--r-- 1 monadayzek staff 0 4 okt. 15:55 /Users/monadayzek/newfile  
-rw----- 1 monadayzek staff 0 4 okt. 16:02 /Users/monadayzek/privatefile  
drw-rw-rw- 2 monadayzek staff 64 4 okt. 16:04 /Users/monadayzek/publicdir  
-rw-r--r-- 1 monadayzek staff 0 4 okt. 16:02 /Users/monadayzek/publicfile
```

б. выведите полную информацию о объектах домашнего каталога пользователя, имена которых начинаются с точки.

```
monadayzek@MacBook-Air-Monad ~ % ls -ld ~/?.?* ~/?.?*  
drwxr-xr-x@ 17 monadayzek staff 544 17 сент. 15:57 /Users/monadayzek/.anydesk  
drwxr-xr-x@ 17 monadayzek staff 544 17 сент. 15:57 /Users/monadayzek/.anydesk  
drwxr-xr-x@ 4 monadayzek staff 128 27 авг. 23:24 /Users/monadayzek/.aspnet  
drwxr-xr-x@ 4 monadayzek staff 128 27 авг. 23:24 /Users/monadayzek/.aspnet  
-rw----- 1 monadayzek staff 584 3 окт. 21:30 /Users/monadayzek/.bash_h  
istory  
-rw----- 1 monadayzek staff 584 3 окт. 21:30 /Users/monadayzek/.bash_h  
istory  
-rw-r--r-- 1 monadayzek staff 420 8 авг. 01:36 /Users/monadayzek/.bash_p  
rofile  
-rw-r--r-- 1 monadayzek staff 420 8 авг. 01:36 /Users/monadayzek/.bash_p  
rofile  
-rw-r--r-- 1 monadayzek staff 118 8 авг. 01:36 /Users/monadayzek/.bashrc  
-rw-r--r-- 1 monadayzek staff 118 8 авг. 01:36 /Users/monadayzek/.bashrc  
drwxr-xr-x@ 3 monadayzek staff 96 31 июля 22:08 /Users/monadayzek/.bundle  
drwxr-xr-x@ 3 monadayzek staff 96 31 июля 22:08 /Users/monadayzek/.bundle  
drwxr-xr-x 4 monadayzek staff 128 2 окт. 00:14 /Users/monadayzek/.cache  
drwxr-xr-x 4 monadayzek staff 128 2 окт. 00:14 /Users/monadayzek/.cache  
-r----- 1 monadayzek staff 7 16 июля 23:58 /Users/monadayzek/.CFUser  
TextEncoding  
-r----- 1 monadayzek staff 7 16 июля 23:58 /Users/monadayzek/.CFUser  
TextEncoding  
drwxr-xr-x 6 monadayzek staff 192 4 сент. 12:56 /Users/monadayzek/.colima  
drwxr-xr-x 6 monadayzek staff 192 4 сент. 12:56 /Users/monadayzek/.colima  
drwxr-xr-x@ 9 monadayzek staff 288 13 сент. 18:47 /Users/monadayzek/.config  
drwxr-xr-x@ 9 monadayzek staff 288 13 сент. 18:47 /Users/monadayzek/.config  
drwxr-xr-x@ 5 monadayzek staff 160 31 авг. 02:40 /Users/monadayzek/.cursor  
drwxr-xr-x@ 5 monadayzek staff 160 31 авг. 02:40 /Users/monadayzek/.cursor  
drwxr-xr-x 7 monadayzek staff 224 4 сент. 12:56 /Users/monadayzek/.docker  
drwxr-xr-x@ 20 monadayzek staff 224 4 сент. 12:56 /Users/monadayzek/.docker  
drwxr-xr-x@ 20 monadayzek staff 640 29 авг. 11:59 /Users/monadayzek/.dotnet  
drwxr-xr-x@ 20 monadayzek staff 640 29 авг. 11:59 /Users/monadayzek/.dotnet  
-rw-r--r--@ 1 monadayzek staff 61 9 авг. 18:07 /Users/monadayzek/.dotty_
```

с. выведите полную информацию о объектах домашнего каталога пользователя, имеющих длину имени ровно пять символов;

```
monadavzeyk@MacBook-Air-Monad ~ % ls -ld ~/????? (N)
```

```
drwx-----+ 5 monadayzek  staff   160  20 авг.  00:57 /Users/monadayzek/Music  
-rw-r--r--  1 monadayzek  staff  2838  4 окт.  16:38 /Users/monadayzek/users
```

- 
- 
- d. удалите подкаталоги домашнего каталога пользователя, заканчивающихся символами **dir** и файлы содержащие в своем названии подстроку **file**:

Боюсь, но идея такая:

echo rm -ir -- /\*dir(/N) : / - директории, -r – рекурсивно, -i (подтв.) -- если имена с -

rm -i -- /\*file\*(.N)

---

---

---

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:



# Лабораторная работа 7. Инструментальные средства UNIX

## Упражнение 7.1. Упаковщики и архиваторы UNIX

- При помощи команд упаковки **compress**, **uncompress**, **gzip**, **gunzip**, **bzip2**, **lzma** упакуйте один и тот же файл (например, **/usr/share/dict/american-english**, скопировав предварительно в домашний каталог) несколько раз. Примечание 1: если файл для сжатия отсутствует, то скачайте любой словарь для проверки орфографии из сети интернет. Примечание 2: если отсутствуют команды упаковки установите их дополнительно через менеджер управления пакетами.

Сравните скорость и качество упаковки:

```
monadazek@MacBook-Air-Monad words % ls -lh dict.txt*
-rw-r--r-- 1 monadazek wheel 4,6M 11 окт. 12:51 dict.txt
-rw-r--r-- 1 monadazek wheel 1,6M 11 окт. 12:51 dict.txt.bz2
-rw-r--r-- 1 monadazek wheel 1,4M 11 окт. 12:51 dict.txt.gz
-rw-r--r-- 1 monadazek wheel 1,2M 11 окт. 12:51 dict.txt.xz
-rw-r--r-- 1 monadazek wheel 2,0M 11 окт. 12:51 dict.txt.Z
```

brew install xz

---

```
compress dict.txt | dict.txt.Z | LZW / быстро
gzip dict.txt | dict.txt.gz | LZ77 + Huffman / опт
bzip2 dict.txt | dict.txt.bz2 | BW + Huffman / медленно
xz dict.txt | dict.txt.xz | LZMA2 / медленно
```

---

```
ls -lh ~/dict.txt*
// time померить
```

---

- При помощи команды архивации **tar** выполните несколько операций архивирования:
  - создайте архив **dotfiles.tar** содержащий файлы **.bashrc**, **.bash\_profile**, **.bash\_history**, сравните размер архива и суммарный размер файлов:

```
monadazek@MacBook-Air-Monad / % ls -lh ~/.bashrc ~/.bash_profile ~/.bash_
y
-rw----- 1 monadazek staff 605B 6 окт. 14:11 /Users/monadazek/.ba
story
-rw-r--r-- 1 monadazek staff 420B 8 авг. 01:36 /Users/monadazek/.ba
ofile
-rw-r--r-- 1 monadazek staff 118B 8 авг. 01:36 /Users/monadazek/.ba
monadazek@MacBook-Air-Monad / % du -ch ~./.ba
y
4,0K /Users/monadazek/.bashrc
4,0K /Users/monadazek/.bash_profile
4,0K /Users/monadazek/.bash_history
12K total
```

---

```
tar -cvf ~/dotfiles.tar ~/.bashrc ~/.bash_profile ~/.bash_history | create verbose (показать добавления) f – название архива
du -h ~/dotfiles.tar – показать размер + normally
ls -lh ~/dotfiles.tar
```

---

- разархивируйте полученный архив **dotfiles.tar** в каталог **dotfiles**, сравните атрибуты (разрешения, владельцев, время создания) исходных файлов, и файлов из архива:

```
monadayzek@MacBook-Air-Monad / % ls -l ~/.bashrc ~/dotfiles/ ~/.bashrc
-rw-r--r--  1 monadayzek  staff  118  8 авг.  01:36 /Users/monadayzek/.bashrc
-rw-r--r--  1 monadayzek  staff  118  8 авг.  01:36 /Users/monadayzek/.bashrc

/Users/monadayzek/dotfiles/:
total 0
drwxr-xr-x  3 monadayzek  staff  96 11 окт. 13:03 Users
monadayzek@MacBook-Air-Monad / %
```

mkdir ~/dotfiles

tar -xvf ~/dotfiles.tar -C ~/dotfiles | x- распаковка, v – показать, f – файл архива, C куда распаковать  
ls -l ~/bashrc ~/dotfiles/.bashrc

- c. создайте упакованный архив (непосредственно командой **tar**) dotfiles.tar.gz, сравните его размер с размером неупакованного архива dotfiles.tar и суммарным размером файлов архива:

```
tar: Removing leading '//' from member names
a Users/monadayzek/.bashrc
a Users/monadayzek/.bash_profile
a Users/monadayzek/.bash_history
-rw-r--r--  1 monadayzek  staff   4,5K 11 окт. 13:00 /Users/monadayzek/dotfiles.tar
-rw-r--r--  1 monadayzek  staff   752B 11 окт. 13:05 /Users/monadayzek/dotfiles.tar.gz
monadayzek@MacBook-Air-Monad: ~%
```

tar -czvf ~/dotfiles.tar.gz ~/.bashrc ~/.bash\_profile ~/.bash\_history | c – создать, z – gzip, v- процесс, f – имя файла  
ls -lh ~/dotfiles.tar ~/dotfiles.tar.gz

- d. разархивируйте упакованный архив dotfiles.tar.gz в каталог dotfiles-:

```
drwxr-xr-x  3 501  20  96 11 окт. 13:03 Users
```

mkdir ~/dotfiles-

tar -xzvf ~/dotfiles.tar.gz -C ~/dotfiles- | тоже самое но распаковка

---

---

---

## Упражнение 7.2. Текстовые редакторы UNIX

- При помощи команды **vimtutor ru** ознакомьтесь с принципами работы в редакторе **vim** и зафиксируйте основные команды:
  - команды посимвольного перемещения курсора:

h <-, j ->, k ^, l вниз

---

b. команды пословного перемещения курсора:

w, b, e – начало / конец слова

---

c. команды постраничного листания:

ctrl + f, ctrl + b (вперед назад на 1 стр.)

---

d. команды поиска текста:

/pattern, ?pattern (вперед, назад)

---

e. команды повторения предыдущего поиска:

n, N (повтор в том же направлении, обратное направление)

---

f. команды вставки текста:

i,a,o,O - (вставить перед курсором, после, новая строка ниже, новая строка выше)

---

g. команды удаления текста:

x (char), dd (string), dw (word)

---

---

---

h. команды захвата текста:

yy (string), yw (word) :

10 строк

10yy , 10Y, y10j – 10 строк вниз + текущая, 10 yw – 10 слов в визуальном V -> 9j

---

i. команды вставки захваченного текста:

p (после), P(перед)

---

---

j. команды записи текста в файл:

:w

---

---

k. команды чтения файла:

:r file

---

---

l. команды выхода из редактора:

:q, :q!, :wq

---

---

2. При помощи редактора **vim** отредактируйте файл .plan, куда внесите информацию о себе.

```
monad@MacBook-Air-Monad ~ % cat ~/.plan
I love Haskell
```

vim ~/.plan -> :wq

## Упражнение 7.3. Файловый менеджер UNIX – Midnight Commander

При помощи файлового менеджера **mc** выполните пункты [лаб. 4](#) и [лаб. 5](#).

Левая панель	Файл	Команда	Настройки	Правая панель
<-- ~		[^>]	<-- /bin	[.^>]
.и Имя	Размер	Дата правки	.и Имя	Размер
./..	-BВЕРХ-	окт. 4 15:42	./..	-BВЕРХ-
/.ServiceHub		96 авг. 30 04:21	*	101456 сент. 25 10:03
/.Trash	64	окт. 11 02:15	*bash	1310208 сент. 25 10:03
/.anydesk	544	сент. 17 15:57	*cat	118992 сент. 25 10:03
/.aspnet	128	авг. 27 23:24	*chmod	120656 сент. 25 10:03
/.bundle	96	июля 31 22:08	*cp	153368 сент. 25 10:03
/.cache	128	окт. 11 13:16	*csh	1091936 сент. 25 10:03
/.colima	192	сент. 4 12:56	*dash	274272 сент. 25 10:03
/.config	320	окт. 11 13:16	*date	135392 сент. 25 10:03
/.cursor	160	авг. 31 02:40	*dd	152576 сент. 25 10:03
/.docker	224	сент. 4 12:56	*df	119152 сент. 25 10:03
/.dotnet	640	авг. 29 11:59	*echo	161136 сент. 25 10:03
/.eclipse	160	авг. 20 12:52	*ed	202528 сент. 25 10:03
/.emacs.d	128	авг. 4 06:08	*expr	182900 сент. 25 10:03
/.gem	128	июля 31 21:50	*hostname	101184 сент. 25 10:03
-BВЕРХ-			*bash	
		54G / 228G (23%)		54G / 228G (23%)

## **Упражнение 7.4. Подсистемы отложенных и периодических заданий.**

The screenshot shows a terminal window with the following content:

Левая панель      Файл      Команда      Настройки      Правая панель

Имя	Размер	Дата правки
.и	160	сент. 27 23:48
/Documents		
/Download		
/Library		
/Movies		
/Music		
/Pictures		
/Public		
/dotfiles		
/dotfiles		
/etc		
/groupsub		
/publicdi		
.CFUser~		
.DS_Stor		
.RData		

Дерево каталогов

```
/
  .nofollow
  .resolve
  .vol
  Applications
  Library
  System
  Users
  Volumes
```

Справа от дерева каталогов расположена таблица истории команд:

Имя	Размер	Дата правки
./..	-ВВЕРХ-	окт. 4 15:42
		30 04:21
		11 02:15
		17 15:57
		27 23:24
		31 22:08
		11 13:16
		4 12:56
		11 13:16
		31 02:40
		4 12:56
		29 11:59
		20 12:52
		4 06:08
		31 21:50

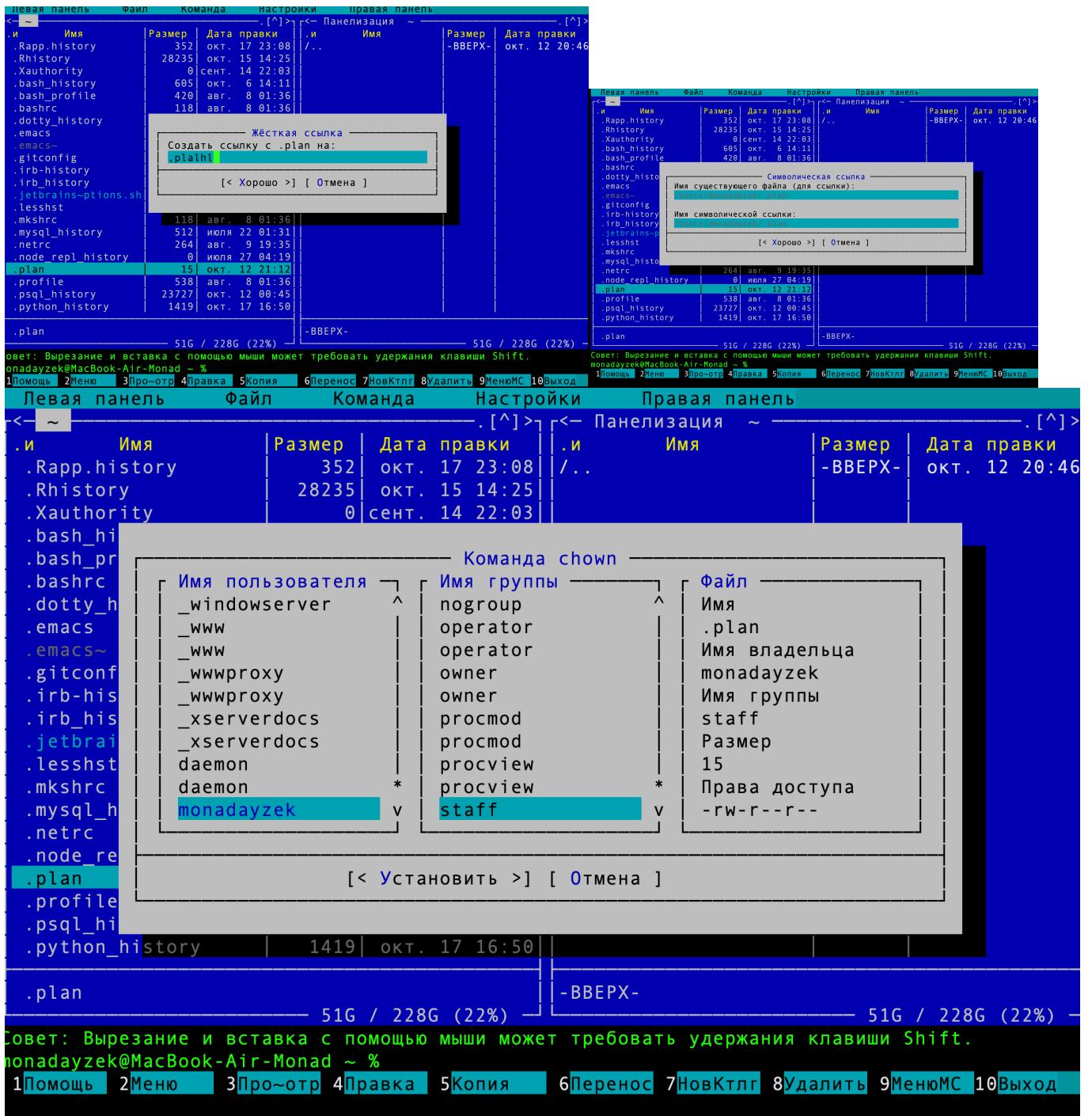
Внизу экрана видны сообщения о свободном и занятом пространстве на диске: 54G / 228G (23%)

Совет: Используйте M-р и M-п для доступа к истории команд.

monadzyek@MacBook-Air-Monad ~ %

Информация  
Midnight Commander 4.8.33

Файл: ..  
Положение: 100000Fh:4335h  
Права: drwxr-xr-x (0755)  
Атрибуты: недоступно  
Ссылок: 8  
Владелец: root/admin  
Размер: 256(0 блоков)  
Изменён: окт. 12 20:46  
Модифицирован: окт. 12 20:46  
Доступ: окт. 15 11:39  
ФС: /  
Устройство: /dev/disk3s1s1  
Тип: apfs  
Своб. место: 51G/228G (22%)  
Своб. узлов: 534976360 / 535423134 (99%)



brew install mc

Информация

### 1. При помощи команд **at**, **atq**, **atrm**:

- a. установите отложенное (на пять минут вперед) задание для создания упакованного архива в домашнем каталоге с именем autoarchive-backup.tgz, содержимого которого включает весь домашний каталог пользователя:

```
monadazek@MacBook-Air-Monad ~ % echo "tar -czf ~/autoarchive-backup.tgz ~
now + 5 minutes
job 2 at Sat Oct 11 13:27:00 2025"
```

echo "tar -czf ~/autoarchive-backup.tgz ~" | at now + 5 minutes (echo передаст в at, gzip архив весь домашний каталог, через 5 мин.)

- 
- 
- b. удостоверьтесь, что отложенное задание создано, проследите за изменениями:

```
monadayzek@MacBook-Air-Monad ~ % atq
1      Sat Oct 11 13:27:00 2025
2      Sat Oct 11 13:27:00 2025
```

atq

---

---

---

- c. дождитесь времени выполнения отложенного задания, удостоверьтесь в корректности его выполнения:

```
autoarchive-backup.tgz
```

- 
- 
- 
- d. создайте любое отложенное задание, затем удалите его, проследите за сообщениями:

```
job 3 at Sat Oct 11 13:26:00 2025
1      Sat Oct 11 13:27:00 2025
3      Sat Oct 11 13:26:00 2025
2      Sat Oct 11 13:27:00 2025
monadayzek@MacBook-Air-Monad ~ % atrm 3
monadayzek@MacBook-Air-Monad ~ % atq
1      Sat Oct 11 13:27:00 2025
2      Sat Oct 11 13:27:00 2025
```

echo 'echo "Hello, from at!" > ~/hello.txt' | at now + 1 minute - | внешний передает в at, внутренний команда.

atq

atrm 3

---

---

---

- e. удостоверьтесь, что отложенное задание удалено, проследите за изменениями:

выше

---

---

---

2. При помощи команды **crontab**:

- a. установите периодическое (с шаблоном времени выполнения каждые пять минут) задание для создания упакованного архива в домашнем каталоге с именем autoarchive-backup.tgz, содержимого которого включает весь домашний каталог пользователя:

```
monadayzek@MacBook-Air-Monad ~ % crontab -e  
crontab: no crontab for monadayzek - using an empty one  
crontab: installing new crontab  
monadayzek@MacBook-Air-Monad ~ %
```

crontab -e

5 5 \* \* \* tar -czf ~/autoarchive-backup.tgz ~ | -e редактирование в редакторе (у меня vim)  
\* - minute, \* - hour, \* - day, \* - month, \* week (0-6).

- запуск на 5 минуте каждого часа. 00:05, 01:05 ...  
- 0 5 \* \* \* - каждый день в 5 часов

- b. удостоверьтесь, что периодическое задание создано, проследите за сообщениями:

```
crontab: installing new crontab
```

```
monadayzek@MacBook-Air-Monad ~ % crontab -l  
*/5 * * * * tar -czf ~/autoarchive-backup.tgz ~  
monadayzek@MacBook-Air-Monad ~ %
```

- 
- 
- 
- c. дождитесь времени выполнения периодического задания, удостоверьтесь в корректности его периодического выполнения:

выше

---

---

---

- d. удалите установленное периодическое задание, проследите за сообщениями:

crontab -r

- e. удостоверьтесь, что периодическое задание удалено, проследите за изменениями:

```
monadayzek@MacBook-Air-Monad ~ % crontab -r
monadayzek@MacBook-Air-Monad ~ % crontab -l
crontab: no crontab for monadayzek
monadayzek@MacBook-Air-Monad ~ % █
```

crontab -l

---

---

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:



# Лабораторная работа 8. Процессы UNIX

## Упражнение 8.1. Типы процессов

1. При помощи команды **ps** aux найдите среди всех процессов операционной системы:
- системные процессы

```
monadayzek@MacBook-Air-Monad ~ % ps aux
USER          PID %CPU %MEM      VSZ      RSS   TT  STAT STARTED           TIME COMMAND
D
monadayzek    56320 17,8  0,5 436634704  43696    ??  Ss   7:12   0:27.02 /Appl
monadayzek    1261 10,2  0,3 435496464  23648    ??  S   29сент.25 24:42.89 /us
r/
monadayzek    1067  9,4  0,1 435355168  11856    ??  S   29сент.25 46:08.12 /us
r/
monadayzek    56403  8,8  0,4 411465872  37248    ??  Ss   7:13   0:01.76 /Syst
root         581  5,9  0,1 435346800   9424    ??  Ss   29сент.25 12:36.46 /us
r/
 _windowserver  614  5,6  1,4 437166160 118048    ??  Ss   29сент.25 1181:07.13 /S
y
```

- Все пользователи, и – имя пользователя, сри, гам, time. x – показать без управляющего терминала

ps aux | grep "root"

---

---

- Демоны

```
monadayzek@MacBook-Air-Monad ~ % ps axo "pid= ppid= user= tty= stat= comm=" | awk '$4 ~ /^$/'
+$/'

      1      0 root      ??      Ss   /sbin/launchd
     322      1 root      ??      Ss   /usr/libexec/logd
     324      1 root      ??      Ss   /usr/libexec/UserEventAgent
     326      1 root      ??      Ss   /System/Library/Frameworks/CoreServices.framework/Versions/A/Frameworks/FSEvents.framework/Versions/A/Support/fsevents
     327      1 root      ??      Ss   /System/Library/PrivateFrameworks/MediaRemote.framework/Support/mediaremoted
     330      1 root      ??      Ss   /usr/sbin/systemstats
     334      1 root      ??      Ss   /usr/libexec/configd
     336      1 root      ??      Ss   /System/Library/CoreServices/powerd.bundle/powerd
     337      1 root      ??      Ss   /usr/libexec/IOMFB_bics_daemon
     345      1 _corespeechd      ??      Ss   /System/Library/PrivateFrameworks/CoreSpeech.framework/corespeechd_system
     347      1 root      ??      Ss   /usr/libexec/watchdogd
     351      1 root      ??      Ss   /System/Library/Frameworks/CoreServices.framework/Versions/A/Support/mds
     353      1 root      ??      Ss   /usr/libexec/kernelmanagerd
     354      1 root      ??      Ss   /usr/libexec/diskarbitrationd
     357      1 root      ??      Ss   /usr/libexec/coreduetd
     358      1 root      ??      Ss   /usr/sbin/syslogd
```

ps aux | grep -v TTY – не только демоны

только демоны – ps axo "pid= ppid= user= tty= stat= comm=" | awk '\$4 ~ /^\$/'

'\$4 ~ /^\$/' — в 4-м столбце (TTY) строка состоит из ? то есть значит нет TTY (демон/фон).

---

---

- пользовательские процессы

```
monadayzek@MacBook-Air-Monad ~ % ps -U "$USER" -u "$USER" -x -o "pid= ppid= user= tty= stat= comm="

 397 1 monadayzek ?? S /System/Library/CoreServices/loginwindow.app/Contents/
 578 1 monadayzek ?? S /usr/sbin/distnoted
 579 1 monadayzek ?? S /usr/sbin/cfprefsd
 584 1 monadayzek ?? S /usr/libexec/UserEventAgent
 589 1 monadayzek ?? S /usr/libexec/knowledge-agent
 590 1 monadayzek ?? S /System/Library/CoreServices/CoreServicesUIAgent.app/Content
 591 1 monadayzek ?? S /usr/sbin/universalaccesssd
 592 1 monadayzek ?? S /usr/libexec/pboard
 593 1 monadayzek ?? S /usr/libexec/containermanagerd
 594 1 monadayzek ?? S /System/Library/CoreServices/WindowManager.app/Content
 595 1 monadayzek ?? S /System/Library/PrivateFrameworks/BiomeStreams.framework
 596 1 monadayzek ?? S /usr/libexec/lspd
 598 1 monadayzek ?? S /System/Library/PrivateFrameworks/BackgroundTaskManager
 599 1 monadayzek ?? S /usr/libexec/linkd
 601 1 monadayzek ?? S /usr/libexec/secd
 605 1 monadayzek ?? S /System/Library/Frameworks/ApplicationServices.framework
 606 1 monadayzek ?? S /System/Library/CoreServices/talagentd
 608 1 monadayzek ?? S /usr/libexec/nsurlsessiond
 609 1 monadayzek ?? S /usr/libexec/rapportd
 613 1 monadayzek ?? S /System/Library/PrivateFrameworks/CoreDuetContext.framework
```

ps -u \$USER -  
ps -U "\$USER" -u "\$USER" -x -o "pid= ppid= user= tty= stat= comm=" (без tty что бы не видеть фоновые + -U для эфф.  
UID и -u для реального UID)

СКРИН

---

---

2. При помощи команды **ps aux** найдите и зафиксируйте все процессы, порожденные от процесса init:

```
monadayzek@MacBook-Air-Monad ~ % pstree
+= 00001 root /sbin/launchd
|-+ 00221 monadayzek /System/Cryptexes/App/System/Library/CoreServices/SafariSuppo
| |-+ 00250 monadayzek /usr/libexec/networkserviceproxy
| |-+ 00541 root /usr/libexec/logd
| |-+ 00543 root /usr/libexec/UserEventAgent (System)
| |-+ 00545 root /System/Library/Frameworks/CoreServices.framework/Versions/A/Framew
| |-+ 00546 root /System/Library/PrivateFrameworks/MediaRemote.framework/Support/med
| +-+ 00549 root /usr/sbin/systemstats --daemon
| \--- 00769 root /usr/sbin/systemstats --logger-helper /private/var/db/systemstats
```

В macos – нет ps aux, но есть через брю pstree

---

---

3. Проследите и отметьте всю цепочку порождаемых процессов от процесса init до процесса ps, с выводом которого вы работаете:

```
monadayzek@MacBook-Air-Monad ~ % pstree | grep -A 5 launchd
```

```
-+ 00001 root /sbin/launchd
|-+ 00221 monadayzek /System/Cryptexes/App/System/Library/CoreServices/SafariSuppor
t.bundle/Contents/MacOS/SafariBookmarksSyncAgent
|-+ 00250 monadayzek /usr/libexec/networkserviceproxy
|-+ 00541 root /usr/libexec/logd
|-+ 00543 root /usr/libexec/UserEventAgent (System)
|-+ 00545 root /System/Library/Frameworks/CoreServices.framework/Versions/A/Frameworks/FSEvents.framework/Versions/A/Support/fsevents
--+
|--- 00609 root /usr/libexec/corebrightnessd --launchd
|--- 00610 root /usr/libexec/AirPlayXPCHelper
```

-A – after 5

## Упражнение 8.2. Атрибуты процессов

- Запустите программу **top**. Нажмите клавишу **h** с основными принципами использования этой программы:

```
monadazek@MacBook-Air-Monad ~ % top -h
top usage: top
          [-a | -d | -e | -c <mode>]
          [-F | -f]
          [-h]
          [-i <interval>]
          [-l <samples>]
          [-ncols <columns>]
          [-o <key>] [-O <secondaryKey>]
                      keys: pid (default), command, cpu, cpu_me, cpu_others, csw,
                           time, threads, ports, mregion, mem, rprvt, purg, vsize, vprv
t,
                           kprvt, kshrd, pgrp, ppid, state, uid, wq, faults, cow, user,
                           msgsent, msgrecv, sysbsd, sysmach, pageins, boosts, instrs,
cycles
          [-R | -r]
          [-S]
          [-s <delay>]
          [-n <nprocs>]
          [-stats <key(s)>]
          [-pid <processid>]
          [-user <username>]
          [-U <username>]
          [-u]
          [-W]
```

- 
- 
- 
- Заставьте программу отображать следующие поля: **PID, PPID, NI, UID, PRI, TTY, USER**

```
monadazek@MacBook-Air-Monad ~ % top -l 1 -o pid -stats pid,ppid,uid,user,state,cpu,mem,threads

Processes: 586 total, 3 running, 583 sleeping, 2582 threads
2025/10/18 14:42:13
Load Avg: 1.67, 2.07, 2.33
CPU usage: 7.47% user, 14.95% sys, 77.57% idle
SharedLibs: 502M resident, 108M data, 72M linkedit.
MemRegions: 95617 total, 1524M resident, 166M private, 1208M shared.
PhysMem: 7551M used (1914M wired, 2236M compressor), 65M unused.
VM: 237T vsize, 5176M framework vsize, 68240(0) swapins, 112404(0) swapouts.
Networks: packets: 32634179/40G in, 7341518/2060M out.
Disks: 10558534/300G read, 3298077/71G written.

      PID      PPID     UID   USER           STATE    %CPU   MEM   #TH
  28829    28513     0   root    running    0.0   4240K  1/1
  28828        1    501 monadazek   sleeping   0.0   2688K  3
```

Можно вывести только без Ni, PRI, TTY. Либо вывести ps

---

```
ps axo "pid=PID ppid=PPID nice=NI uid=UID pri=PRI tty=TTY user=USER"
```

---

```
top -l 1 -o pid -stats pid,ppid,uid,user,state,cpu,mem,threads
```

---

```
-l 1 – даст 1-у выборку и выйдет иначе будет бесконечно -o сортирует по pid
```

- Измените атрибут **NICE** процесса top, проследите за изменившимся приоритетом процесса.

```
monadazek@MacBook-Air-Monad ~ % ps -o pid,ni,pri,comm -p 1261
 PID NI PRI COMM
 1261 10  52 /usr/libexec/avconferenced
```

-о – указать что выводить

Ni – nice / приоритет пользов.

Pri / приоритет ядра который дает nice + система

Comm – имя процесса

renice N -p PID

понизить можно повысить через SUDO - при понижении больше

если создавать то nice -n N command arg

## Упражнение 8.3. Задания и сигналы

1. Запустите программу просмотра любой известной страницы руководства **man** на выполнение на переднем фоне.
2. Завершите полученный в предыдущем пункте процесс нажатием комбинации клавиш **^C** (**SIGINT**). Проследите за реакцией системы:

Man ls fg+

Ни sigint, ни EOF не сработали

---

---

---

3. Запустите программу просмотра любой известной страницы руководства **man** на выполнение на переднем фоне.
4. Завершите полученный в предыдущем пункте процесс нажатием комбинации клавиш **^\** (**SIGQUIT**). Проследите за реакцией системы:

Сработало с sigquit по идеи он прервёт и сделает снимок дампа в core.pid но у меня не получилось увидеть

---

---

---

5. Запустите программу просмотра любой известной страницы руководства **man** на выполнение на переднем фоне.
6. Приостановите полученный в предыдущем пункте процесс нажатием комбинации клавиш **^Z** (**SIGTSTP**). Проследите за сообщениями системы:

Zsh : suspended man ls – то есть заморожен

---

- 
- 
7. При помощи команд **fg**, **bg** поместите приостановленное задание на передний (задний) фон соответственно:

```
monadayzek@MacBook-Air-Monad ~ % bg %1
[1] + continued man ls
monadayzek@MacBook-Air-Monad ~ %
[1] + suspended (tty output) man ls
```

Fg %1  
Bg %1 man / find

---

Потому что **man** использует **less** – а ему нужен TTY для ввода, ядро посыпает ему SIGINT/SIGOUT и шелл покажет stopped. А файнд не интерактивный и без ввода вывода и спокойно работает на фоне

---

8. Последовательно запустите несколько программ просмотра страниц руководства **man** на выполнение, перемещая получаемы процессы на задний фон. Проследите за присвоенными идентификаторами процессов **PID** и заданий **JOBID**:

Man ls &  
Man ps &  
Man top &  
[1]...[3]

---

---

9. При помощи команды **jobs** просмотрите и зафиксируйте список выполняющихся заданий:

```
monadayzek@MacBook-Air-Monad ~ % jobs
[1]      suspended (tty output) man ls
[2]      suspended (tty output) man ls
[3] +     suspended (tty output) man ps
[4] -     suspended (tty output) man top
```

---

---

10. При помощи команды **kill** посыпайте различным заданиям сигналы **SIGINT**, **SIGTERM**, **SIGQUIT**, **SIGKILL**, **SIGSTOP**, **SIGCONT**, проследите за результатом:

```
monadayzek@MacBook-Air-Monad ~ % jobs -l
[1]      57217 suspended (tty output)  man ls
[2]      57253 suspended (tty output)  man ls
[3] +  57254 suspended (tty output)  man ps
[4] -  57255 suspended (tty output)  man top
```

-I – получили пицы

=> kill -SIGINT 57217 / kill -2 – прервать

=> kill -SIGTERM / kill -15 – запрос завершения

=> kill -SIGKILL / kill -9 – убить

=> kill -SIGQUIT – завершить с дампом

=> kill -SIGSTOP / kill -19 – остановить

=> kill SIGCONT / kill -18 – возобновить

---

---

---

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:

# Лабораторная работа 9. Сетевое окружение UNIX

## Упражнение 9.1. Удаленный доступ в систему с помощью протокола SSH

- При помощи команды **ssh** подключитесь к операционной системе соседа:

На макос, сначала нужно дать удаленный доступ. Так как я буду пользоваться только своей машиной, я подключусь к юзерам. Для этого SysSet -> Sharing -> Remote Login (Add users)  
Ssh student@localhost

- При помощи следующих команд: **whoami**, **w**, **who**, **users**, **id**, **date**, **uname**, **tty**, **uptime** получите информацию о **УДАЛЕННОЙ** системе и пользователях и зафиксируйте ее:

```
student@MacBook-Air-Monad / % whoami
student
student@MacBook-Air-Monad / % w
20:37  up 12 days, 21:20, 3 users, load averages: 1,79 2,72 3,00
USER      TTY      FROM          LOGIN@  IDLE WHAT
monadayzek  console   -           29сент. 12days -
monadayzek s000     -           20:25           - ssh
student    s001     localhost  20:34           - w
student@MacBook-Air-Monad / % who
monadayzek      console      29 сент. 23:19
monadayzek      ttys000     12 окт. 20:25
student        ttys001     12 окт. 20:34 (::1)
student@MacBook-Air-Monad / % id
uid=502(student) gid=20(staff) groups=20(staff),12(everyone),61(localaccounts),399(com.apple.access_ssh),704(com.apple.sharepoint.group.4),701(com.apple.sharepoint.group.1),702(com.apple.sharepoint.group.2),703(com.apple.sharepoint.group.3),100(_lpoperator)
student@MacBook-Air-Monad / % date
воскресенье, 12 октября 2025 г. 20:37:23 (MSK)
student@MacBook-Air-Monad / % uname
Darwin
student@MacBook-Air-Monad / % tty
/dev/ttys001
student@MacBook-Air-Monad / % uptime
20:37  up 12 days, 21:21, 3 users, load averages: 1,86 2,68 2,97
student@MacBook-Air-Monad / %
```

- Закончите сеанс работы в операционной системе на машине соседа.

Ctrl + D / exit :

student@MacBook-Air-Monad / % exit

Connection to localhost closed.

- При помощи команды **ssh** запустите команду вывода полной информации о файлах домашнего каталога на **УДАЛЕННОЙ** операционной системе машины соседа, проследите за появляющимися сообщениями:

```
monadayzek@MacBook-Air-Monad ~ % ssh student@localhost "ls -la ~"

(student@localhost) Password:
total 24
drwxr-x---+ 15 student staff 480 30 сент. 16:28 .
drwxr-xr-x  8 root   admin 256 12 окт. 20:31 ..
-rw-----  1 student staff  3 27 сент. 12:53 .CFUserTextEncoding
-rw-----  1 student staff  20 27 сент. 13:43 .lessht
-rw-----  1 student staff 105 27 сент. 13:55 .zsh_history
drwx----- 5 student staff 160 27 сент. 13:55 .zsh_sessions
-rw-r--r--  1 student staff  20 30 сент. 16:28 .zshrc
drwx-----+ 3 student staff  96 27 сент. 12:53 Desktop
drwx-----+ 3 student staff  96 27 сент. 12:53 Documents
drwx-----+ 3 student staff  96 27 сент. 12:53 Downloads
drwx-----+ 30 student staff 960 27 сент. 12:53 Library
drwx-----  3 student staff  96 27 сент. 12:53 Movies
drwx-----+ 3 student staff  96 27 сент. 12:53 Music
drwx-----+ 3 student staff  96 27 сент. 12:53 Pictures
drwxr-xr-x+ 4 student staff 128 27 сент. 12:53 Public
monadayzek@MacBook-Air-Monad ~ %
```

```
monadayzek@MacBook-Air-Monad ~ % ssh student@localhost "ls -la ~"
```

```
(student@localhost) Password:
Could not chdir to home directory /Users/student: Permission denied
total 0
ls: /Users/student: Permission denied
```

Тогда с админа меняем группу :

```
sudo chown -R student:staff /Users/student
```

## Упражнение 9.2. Локальная и сетевая почта

- При помощи команды **mail** пошлите несколько писем различного содержания пользователю **vinnie**, проследите за появляющимися сообщениями:

```
monadayzek@MacBook-Air-Monad ~ % echo 'VinniePuh? Eto ti???' | mail -s 'MonadToVinnie' vinnie

monadayzek@MacBook-Air-Monad ~ % su vinnie
Password:
vinnie@MacBook-Air-Monad monadayzek % mail
No mail for vinnie
vinnie@MacBook-Air-Monad monadayzek %
```

Нужно с homebrew поставить mailutils : brew install mailutils

Так как я использую свою же машину, на маке невозможно отправить локально юзеру почту.  
Из StackOverFlow:

```
sudo launchctl enable system/com.apple.postfix.master
sudo launchctl kickstart -k system/com.apple.postfix.master
Включает службу postfix которая даст такое разрешение
```

Но даже при этом пусто:  
(watch syslog)  
/var/spool/mail

- Прочитайте и удалите прочитанные письма пользователя **vinnie**, проследите за появляющимися сообщениями:

У mail : в окне – h: список, 1 – открыть первое, d 1 – удалить 1-ое, q – выйти

## Упражнение 9.3. Передача файлов с помощью протокола FTP

- При помощи команды **ftp** подключитесь к серверу ftp.yandex.ru (анонимно - учетная запись anonymous), проследите за появляющимися сообщениями:

```
monadazek@MacBook-Air-Monad ~ % lftp -u anonymous ftp.yandex.ru
```

Пароль:

```
lftp anonymous@ftp.yandex.ru:~> █
```

В макос – нет ftp, но есть альтернатива в брю, lftp

- Перейдите в каталог /debian/dists/trixie, проследите за появляющимися сообщениями:

```
monadazek@MacBook-Air-Monad ~ % lftp -u anonymous ftp.yandex.ru
```

Пароль:

```
lftp anonymous@ftp.yandex.ru:~> cd /debian/dists/jessie  
cd: Ошибка доступа: 550 Failed to change directory. (/debian/dists/jessie)  
lftp anonymous@ftp.yandex.ru:/> get ChangeLog  
get: Ошибка доступа: 550 Failed to open file. (ChangeLog)  
lftp anonymous@ftp.yandex.ru:/> █
```

```
monadazek@MacBook-Air-Monad ~ % lftp -u anonymous ftp.gnu.org
```

Пароль:

```
lftp anonymous@ftp.gnu.org:~> cd /gnu  
cd ok, каталог=/gnu  
lftp anonymous@ftp.gnu.org:/gnu> get README  
get: Ошибка доступа: 550 Failed to open file. (README)  
lftp anonymous@ftp.gnu.org:/gnu> ls  
drwxr-xr-x 2 3003 3003 4096 Dec 13 2013 3dldf  
drwxr-xr-x 2 3003 3003 4096 Mar 18 2017 8sync  
-rw-r--r-- 1 3003 65534 1492 Jan 25 2001 =README
```

```
lftp anonymous@ftp.gnu.org:/gnu> get =README
```

1492 байта перемещены за 66 секунд (23 б/с)

```
lftp anonymous@ftp.gnu.org:/gnu> █
```

```
monadazek@MacBook-Air-Monad ~ % ls
```

,plan dotfi

=README Downl

:(((

Попробовал на Debian.org. там завис с Согласование FEAT -> нет соединения

Даже после отключения -e "set ftp:use-feat-no" ничего не помогло, TLS так же грузится бесконечно

GNU.org сработал, скачал README но он там =README. Появился в моей директории

3. Получите с удаленного сервера файл Changelog, проследите за появляющимися сообщениями:

Выше

4. Закончите сеанс работы с командой **ftp**, проследите за появляющимися сообщениями:

Ответ – exit (ctrl + D)

5. При помощи команд **ncftp**, **lftp**, **mc** научитесь пользоваться интерактивными клиентами передачи файлов - повторите, выполненные процедуры из [упр. 10.3.](#)

Левая панель	Файл	Команда	Настройки	Правая панель
<- ~			.[^\>]	<- sftp://student@localhost/ .[^]
.	Имя	Размер	Дата правки	.
/..		-ВВЕРХ-		/..
/.ServiceHub		96	авг. 30 04:21	/.nofollow
/.Trash		64	окт. 11 22:29	/.resolve
/.anydesk		544	сент. 17 15:57	/.vol
/.aspnet		128	авг. 27 23:24	/Applications
/.bundle		96	июля 31 22:08	/Library
/.cache		128	окт. 11 13:16	/System
/.colima		192	сент. 4 12:56	/Users
/.config		352	окт. 12 20:57	/Volumes
/.cursor		160	авг. 31 02:40	/bin
/.docker		224	сент. 4 12:56	/cores
/.dotnet		640	авг. 29 11:59	/dev
/.eclipse		160	авг. 20 12:52	~etc
/.emacs.d		128	авг. 4 06:08	~home
/.gem		128	июля 31 21:50	/opt
/.gradle		352	авг. 20 13:03	/private
/.hawtjni		96	июля 25 19:29	/sbin
/.ipython		96	июля 23 18:14	~tmp
/.ivy2		160	сент. 19 01:16	/usr
/.local		128	июля 31 22:04	~var
/.m2		128	июля 25 20:02	!.VolumeIcon.icns
/.matplotlib		96	авг. 17 17:38	.file
/.mongodb		96	июля 17 01:15	
/.ncftp		256	окт. 12 21:43	
/.nextday		128	сент. 18 01:09	
/.npm		192	авг. 3 04:33	
/.nuget		128	авг. 29 11:59	
/.openjfx		96	июля 23 12:38	
/.oracle_jre_usage		96	июля 22 19:33	
/.p2		160	авг. 20 13:10	
/.pgadmin		320	сент. 15 23:37	
-ВВЕРХ-				-ВВЕРХ-
				ГЗС / ГГГС / ГГГД

Левая панель	Файл	Команда	Настройки	Правая панель
			.[^\>]	<- sftp://student@localhost/ .[^\>]
.	Имя	Размер	Дата правки	и Имя
.node_repl_history		0	июля 27 04:19	/..
.plan		15	окт. 12 21:12	/.nofollow
.profile		538	авг. 8 01:36	/.resolve
.pgsql_history		23727	окт. 12 00:45	/.vol
.python_history		1172	окт. 8 01:18	/Applications
.swp		12288	сент. 15 15:30	/Library
.viminfo		33104	окт. 12 21:12	/System
.zlogin		118	авг. 8 01:36	/Users
.zprofile		392	сент. 27 13:42	/Volumes
.zsh_history		60213	сент. 30 16:26	/bin
.zshrc		559	сент. 30 16:28	/cores
<b>.plan</b>		<b>36</b>	<b>окт. 11 13:12</b>	/dev
=README		1492	янв. 25 2001	~etc
Quantum_computing.ipynb		52382	окт. 11 20:58	~home
autoarchive-backup.tgz		27869	окт. 11 13:35	/.opt
data_sev20251004_170646.txt		0	окт. 4 17:06	/.private
defaultfile		0	окт. 4 15:58	/.sbin
dotfiles.tar		4608	окт. 11 13:00	~tmp
dotfiles.tar.gz		752	окт. 11 13:05	/.usr
errors		65801	окт. 4 16:44	~var
forVinnie		0	окт. 4 15:43	!.VolumeIcon.icns
maskedfile		0	окт. 4 16:00	.file
mbox		2773	окт. 11 21:54	
myfifo		0	сент. 28 17:16	
myfiles		3756	окт. 4 16:47	
newfile		0	окт. 4 15:55	
privatefile		0	окт. 4 16:02	
publicfile		0	окт. 4 16:02	
remote.plan		15	окт. 12 21:21	
users		2838	окт. 4 16:38	
users_upper		2838	окт. 4 16:40	
				36 Б в 1 файле
				53G / 228G (23%)
				-ВВЕРХ-

Совет: Домашняя страница Midnight Commander: <https://www.midnight-commander.org>

Так же в ncftp. Как в lftp

Но lftp лучше потому что поддерживает FTP, FTPS, SFTP, HTTP, HTTPS а ncftp только FTP/FTPS

По mc: F9 -> FTP link -> sftp://student@localhost (в одной мои файлы в другой студента через ssh) -> insert , F5, local

## Упражнение 9.4. Передача файлов с помощью протоколов SCP

- При помощи команды **scp** скопируйте файл .plan из домашнего каталога пользователя student машины соседа в домашний каталог пользователя student, с именем remote.plan, проследите за появляющимися сообщениями:

```
monadazek@MacBook-Air-Monad ~ % scp vinnie@localhost:/Users/vinnie/.plan ~/remote.plan
(vinnie@localhost) Password:
.plan
monadazek@MacBook-Air-Monad ~ % su vinnie
vinnie@MacBook-Air-Monad monadazek % cat remote.plan
Plan of Vinnie
```

- Просмотрите содержимое файла remote.plan (наверху)

## Упражнение 9.5. Синхронизация файлов с помощью протокола RSYNC/SSH

- При помощи команды **rsync** синхронизуйте содержимое каталога bin из домашнего каталога пользователя student машины соседа, используя защищенный транспорт SSH, в домашний каталог пользователя student, с именем remote.bin, проследите за появляющимися сообщениями:

```
monadayzek@MacBook-Air-Monad ~ % rsync -av student@localhost:/Users/student/bin/ ~/remote.bin/
(student@localhost) Password:
Transfer starting: 3 files

sent 20 bytes received 160 bytes 1800000 bytes/sec
total size is 46 speedup is 0,26
```

-a (арх режим), -v (подробно)

+ оно сравнивает файлы источника и приёмника

передаёт только отличающиеся части

может работать локально или по сети (через SSH, RSH, TCP)

2. Просмотрите содержимое полученного каталога remote.bin

```
monadayzek@MacBook-Air-Monad ~ % sudo -u student ls remote.bin
hello.sh      notes.txt
monadayzek@MacBook-Air-Monad ~ % █
```

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:

# Лабораторная работа 10. Система X Window

Примечание: при выполнении лабораторной в среде MacOS X или WSL требуется установить дополнительно X server.

## Упражнение 10.1. Создание собственного настольного окружения

1. Установите оконный менеджер **blackbox**:

```
monadayzek@MacBook-Air-Monad ~ % sudo /opt/local/bin/port install blackbox xterm xeyes xcloc
k xcalc
Password:
---> Computing dependencies for blackbox
---> Cleaning blackbox
---> Computing dependencies for xterm
---> Cleaning xterm
---> Computing dependencies for xeyes
---> Cleaning xeyes
---> Computing dependencies for xclock
---> Cleaning xclock
---> Computing dependencies for xcalc
---> Cleaning xcalc
---> Scanning binaries for linking errors
---> No broken files found.
---> No broken ports found.
```

2. Отредактируйте сценарий командного интерпретатора .xinitrc в домашней директории пользователя *student*, сформировать собственное окружение, путем задания в данном сценарии команд запуска клиентов:

- a. **xsetroot**, для установки цвета **darkslategray** корневого окна:  
В .xinitrc нужно прописать xsetroot -solid darkslategray

- b. **xterm** в правом верхнем углу экрана, цвет символов **white** на фоне **black**:



xterm -fg white -bg black -geometry -0+0 &

c. **xeyes** в левом верхнем углу экрана, размером 100 x 50 пикселов:



`xeyes -fg white -bg black -geometry -0+0 &`

d. **xclock** в правом нижнем углу экрана, размером 50 x 50 пикселов:

`xclock -geometry 50x50-0-0 &`

e. **xcalc** в левом нижнем углу экрана:



`xcalc -geometry +0-0 &`

3. Запустите графическую подсистему X Window, отметьте какое настольное окружение пользователя активизировалось:

В файл .xinitrc записал :

`export LANG=ru_RU.UTF-8  
export LC_ALL=ru_RU.UTF-8`

`xsetroot -solid darkslategray  
xterm -fg white -bg black -geometry -0+0 &  
xeyes -geometry 100x50+0+0 &  
xclock -geometry 50x50-0-0 &  
xcalc -geometry +0-0 &  
exec blackbox`

`/opt/X11/bin/startx`

startx — запускает X-сервер XQuartz и читает ~/.xinitrc; активизируется окружение Blackbox (по последней строке exec blackbox).

## Упражнение 10.2. Настройка настольного окружения Xfce

brew install twm

nano ~/.xinitrc

туда записываем:

xsetroot -solid darkslategray &

xterm -geometry -0+0 -fg white -bg black &

xeyes -geometry 100x50+0+0 &

xclock -geometry 50x50-0-0 &

xcalc -geometry +0-0 &

exec twm

сохранили

перед запуском делаем:

defaults write org.xquartz.X11 app\_to\_run /usr/bin>true

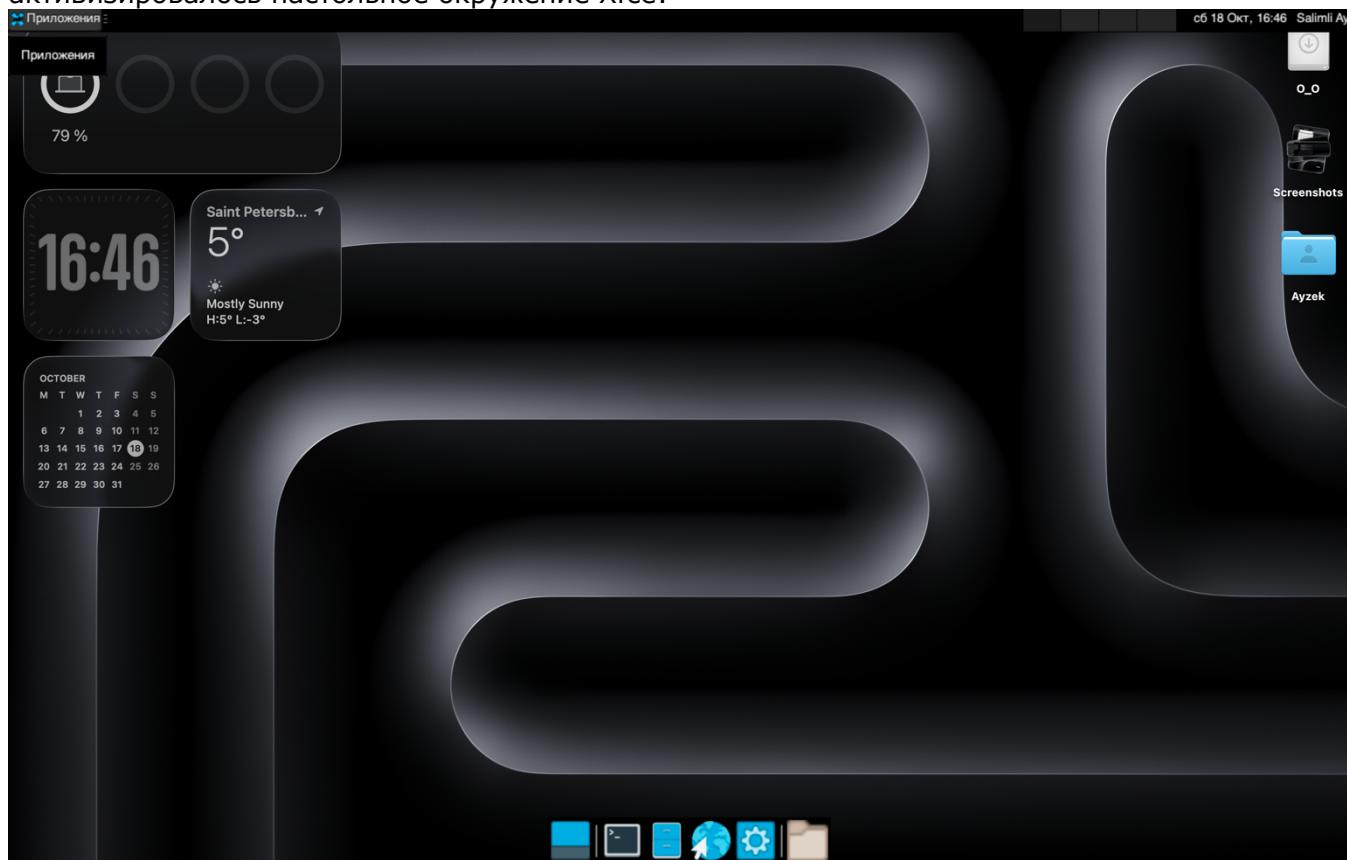
запускаем xquarts

в верхней панельке тыкаем xquarts->настройки->выход-> полноэкранный режим (ставим галку)

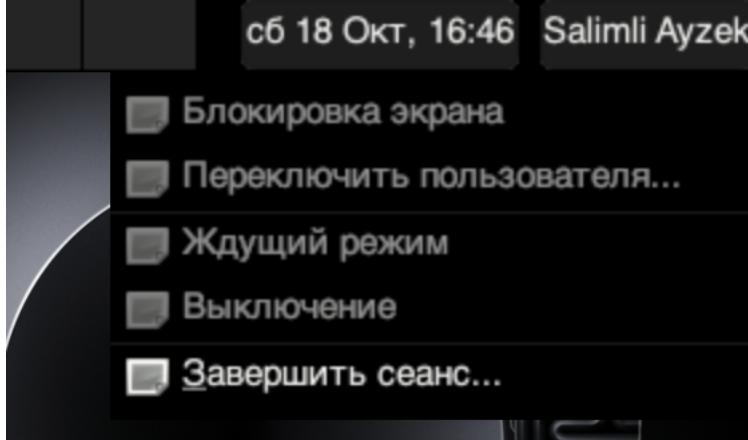
перезапускаем xquarts

в верхней панельке тыкаем xquarts -> переключение полноэкранного режима

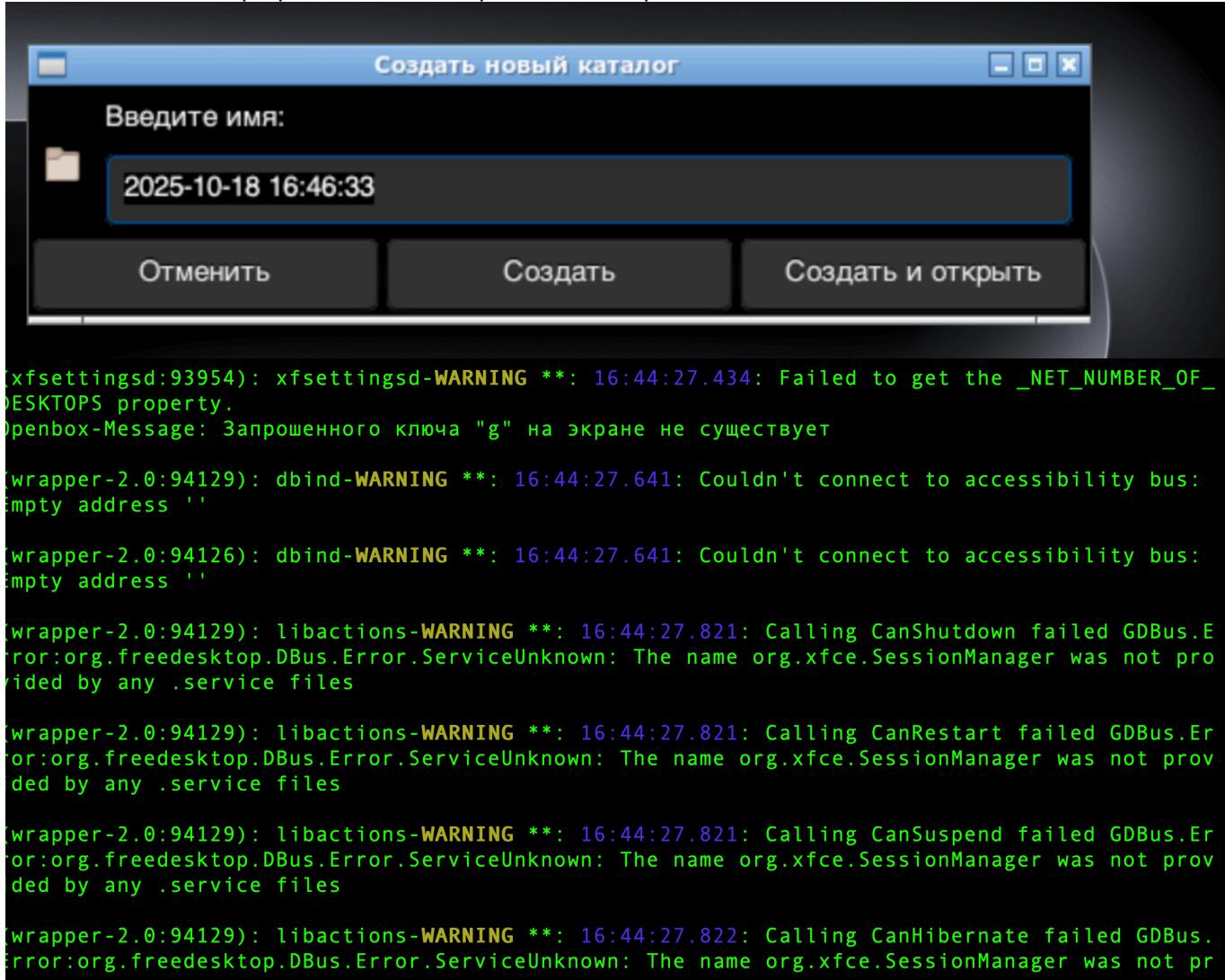
1. Создайте сценарий командного интерпретатора .xinitrc в домашней директории пользователя student таким, чтобы при запуске графической системы X Window всегда активизировалось настольное окружение Xfce:



2. Настройте активизированное настольное окружение Xfce по своему вкусу.



3. Закончите графический сеанс работы в операционной системе.



## Упражнение 10.3. Графический вход в систему

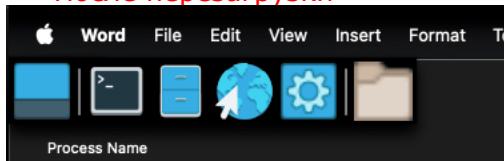
Примечание: в случае если в ОС установлен графический вход в систему по умолчанию, то требуется указать какой используется менеджер дисплеев.

1. Настройте автоматический запуск графической среды с использованием менеджера дисплеев:

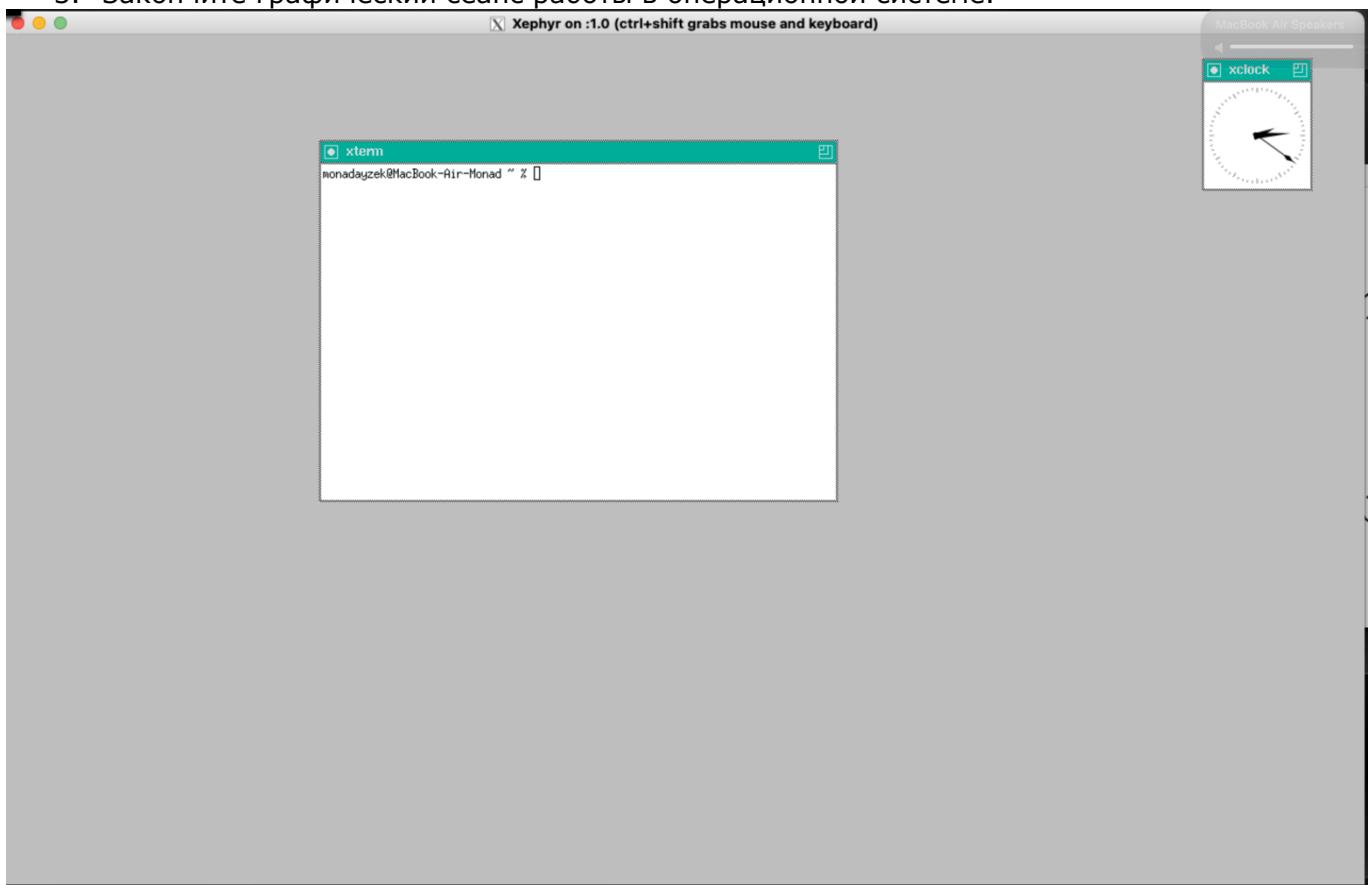
в роли менеджера дисплеев используется launchd (LaunchAgents). Агент org.user.xquartz стартует XQuartz, а org.user.xsession запускает пользовательскую X-сессию (Xfce-панель/настройки/Thunar) под оконным менеджером Openbox.

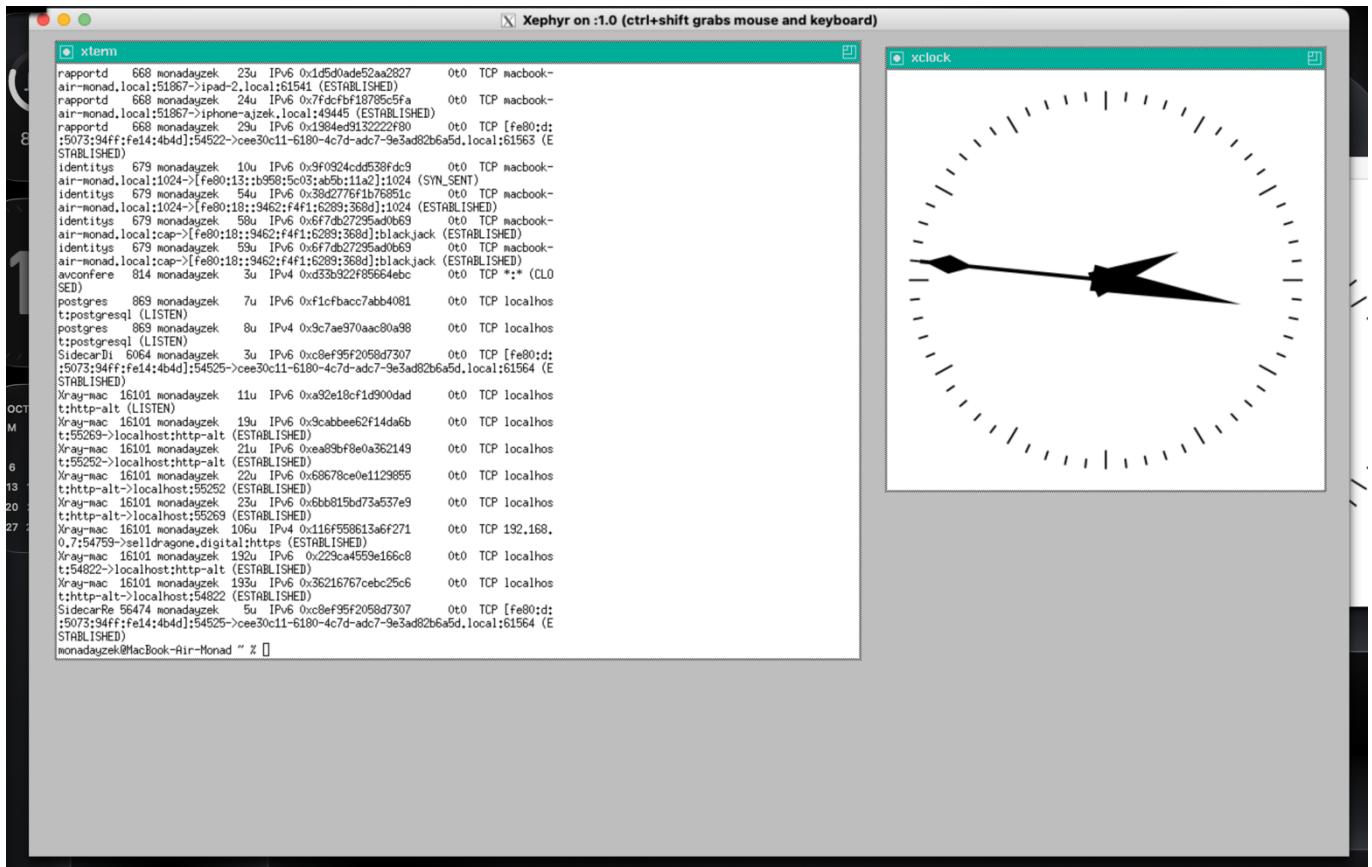
2. Перезагрузите операционную систему. Убедитесь, что доступен графический вход в систему. Используйте настольное окружение Xfce при входе в систему.

**После перезагрузки**



3. Закончите графический сеанс работы в операционной системе.





Openbox --exit

-- попробовать twm  
-- получилось через Xephyr

Лабораторная работа зачтена:  
Дата:  
Подпись преподавателя:

# Лабораторная работа 11. Подсистема печати

## Упражнение 11.1. Средства печати UNIX

1. Установите систему печати **cups**:

```
sudo launchctl kickstart -k system/org.cups.cupsd
```

2. Установите виртуальный драйвер для печати в PDF в систему печати **cups**:

```
mkdir -p ~/CUPS-Output && ippeveprinter -D ~/CUPS-Output -F application/pdf -p 8631 -m "Virtual PDF" -l "Local" "LocalPDF" &
```

создаем папку -р что б не ругался если уже есть папка, ippeveprinter запускает тестовый IPP принтер дльше - D куда класть, -F формат, -р порт, -м Virtual что бы была модель как произвольная строка модель – описание. -l local – что б локально, имя localPDF, & - в фоне.

3. Используя браузер в графической среде, зайдите по адресу <http://127.0.0.1:631> и добавьте принтер с именем **LocalPrinter** использующий драйвер PDF.

```
sudo cupsctl WebInterface=yes && sudo lpadmin -p LocalPrinter -E -v ipp://localhost:8631/ipp/print -m everywhere && sudo cupsenable LocalPrinter && sudo cupsaccept LocalPrinter
```

4. При помощи команд **lpr, lpq, lprm (lp, lpstat, cancel)**:

**Lpr** – будет отправлять файлы в очередь cups, **-P** принтер

- просмотрите состояния принтера с именем **LocalPrinter**:

```
lpstat -p LocalPrinter -l
```

```
monadayzek@MacBook-Air-Monad ~ % lpstat -p LocalPrinter -l
printer LocalPrinter is idle. enabled since суббота, 18 октября 2025 г. 17:31:38
```

- распечатайте любой файл на принтере **LocalPrinter**, проследите за сообщениями:

```
localhost POST /ipp/print
localhost Continue
localhost Get-Printer-Attributes successful-ok
localhost OK
localhost Client closed connection.
monadayzek@MacBook-Air-Monad ~ % lpstat -p LocalPrinter -l
printer LocalPrinter is idle. enabled since суббота, 18 октября 2025 г. 17:30:20
monadayzek@MacBook-Air-Monad ~ % lpr -P LocalPrinter /etc/hosts
monadayzek@MacBook-Air-Monad ~ % localhost POST /ipp/print
localhost Continue
localhost Get-Printer-Attributes successful-ok
localhost OK
localhost POST /ipp/print
localhost Continue
localhost Validate-Job document-format="image/urf"
localhost Validate-Job successful-ok
localhost OK
localhost POST /ipp/print
```

```
lpr -P LocalPrinter /etc/hosts
```

- просмотрите состояния принтера **LocalPrinter**, проследите за сообщениями:

`lpstat -o LocalPrinter && lpstat -p LocalPrinter`

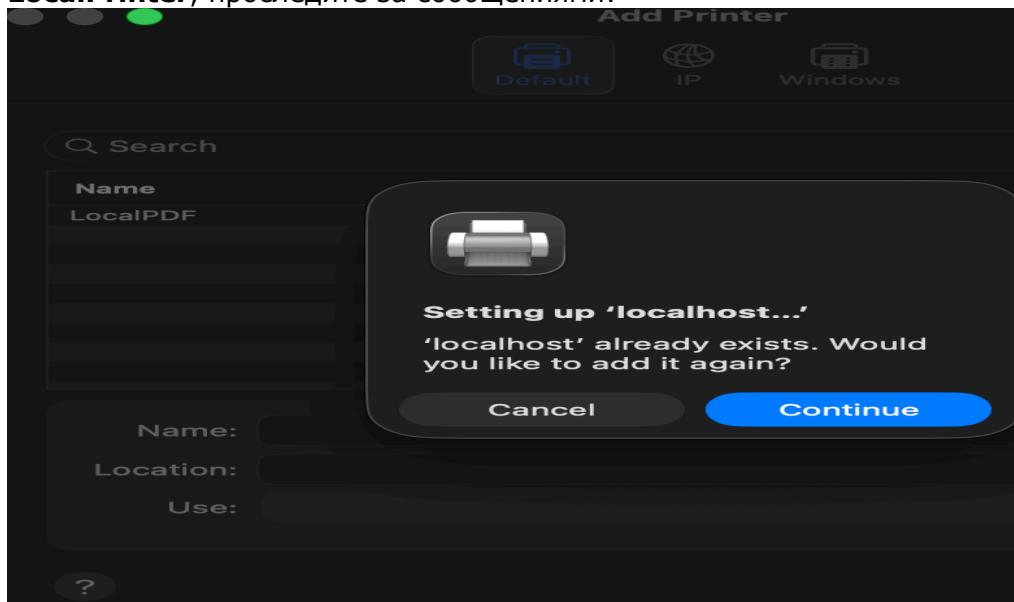
Вывод такой же как в а но ! при печати выдавал **Printing**

-о список в очереди

- d. удалите задание на печать из очереди принтера **LocalPrinter**, проследите за сообщениями:

`lpstat -o LocalPrinter`  
`cancel LocalPrinter-2`

е. распечатайте любую известную страницу руководства **man** на принтере **LocalPrinter**, проследите за сообщениями:



`man -t ls | lpr -P LocalPrinter`

---

но принтер принимает на вход pdf а man это PostScript -> отвечает что формат не тот

---

monadayzek@MacBook-Air-Monad ~ % `man -t ls | lpr -P LocalPrinter lpr:`  
Unsupported document-format "application/postscript".

Но можно с ps2pdf  
`man -t ls | ps2pdf - ~/Desktop/man-ls.pdf`  
Тогда все ок

Лабораторная работа зачтена:  
Дата:  
Подпись преподавателя:



# **Основы программирования на языке командного интерпретатора. Индивидуальное задание**

## **Создание скелета сценария на языке командного интерпретатора BASH**

1. Получите задание на разработку сценария у преподавателя.
2. Создайте файл сценария, назовите его именем, отражающим смысловое назначение. Документируйте назначение вашего сценария в теле самого сценария.
3. Согласно стандарту POSIX.2 придумайте мнемонические модификаторы, которые будут изменять алгоритм работы вашего сценария. Документируйте назначение придуманных модификаторов в теле самого сценария.
4. Создайте скелет сценария выполняющий обработку модификатора -h и выдачу короткой справки по модификаторам и назначению сценария.
5. Дополните скелет сценария, обработкой и проверкой корректности задания остальных модификаторов и/или входных аргументов сценария.

## **Реализация алгоритма сценария на языке командного интерпретатора BASH**

1. Дополните скелет сценария, алгоритмом, выполняющим смысловое назначение сценария.
2. Оттестируйте полученный сценарий в широком спектре входных параметров (как корректных, так и некорректных)

Лабораторная работа зачтена:

Дата:

Подпись преподавателя:

Лаб 13

Нужно написать демон под линукс который бы читал конфигурационный файл, брал оттуда параметры: каталог, время проверки. После чего проверял заданий каталог(рекурсивно), через заданные промежутки времени, на предмет наличия/отсутствия модификации файлов и записывал результаты в файл журнала, предусмотреть обработку сигналов: SIGHUP - для перечитывания конфигурационного файла, и SIGTERM - для контролируемого завершения демона (запись о выходе в файл журнала).

