

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ им. В. И.  
ВЕРНАДСКОГО»  
ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ  
Кафедра компьютерной инженерии и моделирования

**ОТЧЕТ ПО ПРАКТИЧЕСКОМУ ЗАДАНИЮ №5**  
**«Начальная загрузка операционной системы GNU Linux и**  
**периодические процессы»**

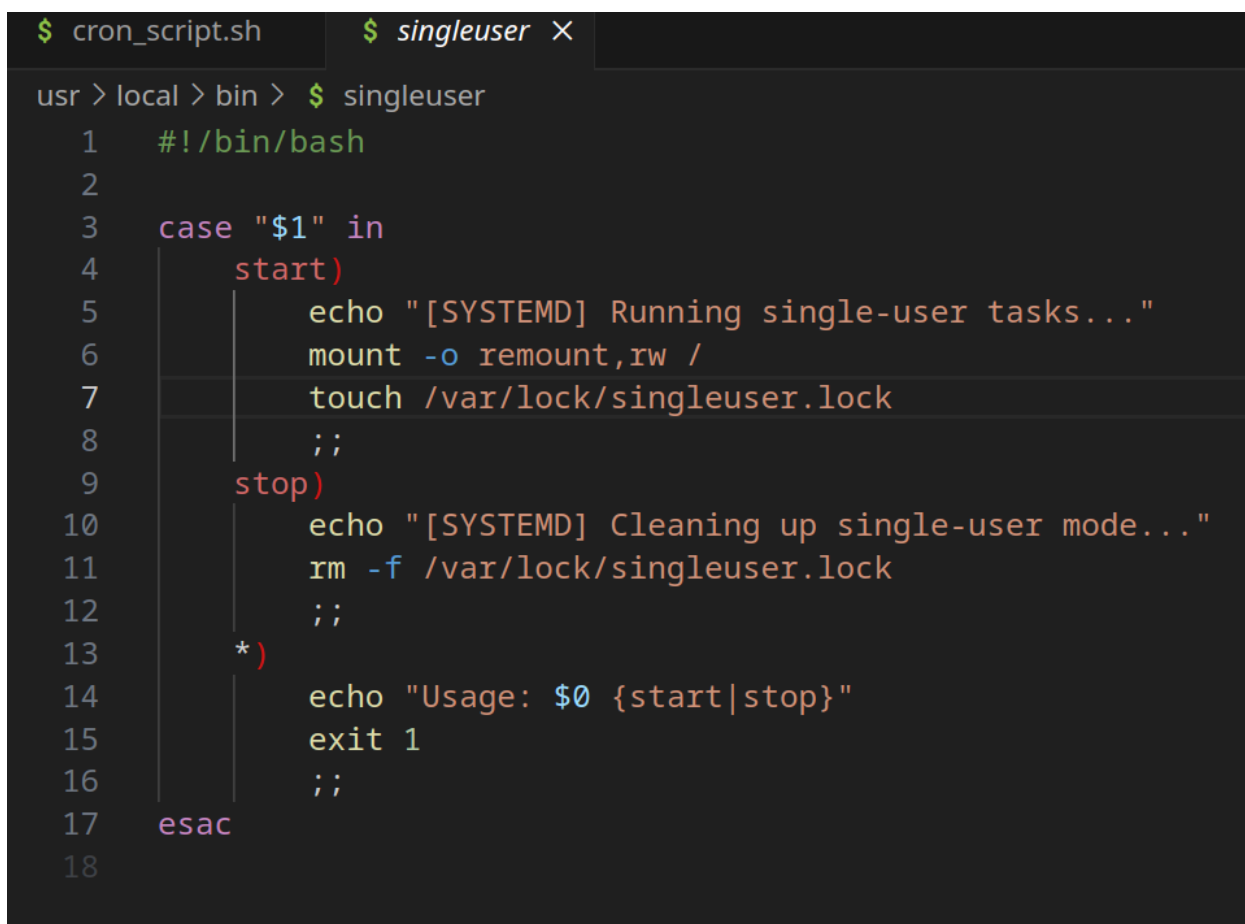
Практическая работа  
по дисциплине «Системное программное обеспечение»  
студента 3 курса группы ИВТ-б-о-222(1)  
Гоголева Виктора Григорьевича

09.03.01 «Направление подготовки»

Симферополь, 2025

## Ход работы

1. Написать стартовый сценарий, который запускается последним при переходе на режим выполнения в однопользовательском режиме. Стартовый сценарий обязан поддерживать параметры остановки и запуска.

A screenshot of a terminal window with a dark background. At the top, there are two tabs: '\$ cron\_script.sh' and '\$ singleuser X'. The terminal shows a directory path 'usr > local > bin >' followed by the command '\$ singleuser'. Below this, the script content is displayed with line numbers 1 through 18. The script is a shell script that uses a 'case' statement to handle 'start', 'stop', and '\*' (usage) commands. It includes commands for echoing status messages, mounting the root filesystem, creating a lock file, and cleaning up. The script ends with 'esac' on line 17.

```
$ cron_script.sh    $ singleuser X
usr > local > bin > $ singleuser
 1  #!/bin/bash
 2
 3  case "$1" in
 4      start)
 5          echo "[SYSTEMD] Running single-user tasks..."
 6          mount -o remount,rw /
 7          touch /var/lock/singleuser.lock
 8          ;;
 9      stop)
10          echo "[SYSTEMD] Cleaning up single-user mode..."
11          rm -f /var/lock/singleuser.lock
12          ;;
13      *)
14          echo "Usage: $0 {start|stop}"
15          exit 1
16          ;;
17  esac
18
```

Рисунок 1 – тело скрипта для сервиса

Добавляем файлу бит на исполнение: `sudo chmod +x /usr/local/bin/singleuser`

```

≡ singleuser.service X
etc > systemd > system > ≡ singleuser.service
1  [Unit]
2  Description=Single-User Mode Custom Script
3  DefaultDependencies=no
4  Before=shutdown.target emergency.target
5
6  [Service]
7  Type=oneshot
8  RemainAfterExit=yes
9  ExecStart=/usr/local/bin/singleuser start
10 ExecStop=/usr/local/bin/singleuser stop
11
12 [Install]
13 WantedBy=rescue.target emergency.target
14

```

/etc/systemd/system/singleuser.servi/etc/systemd/system/singleuser.service

Рисунок 2 – написание конфигурационного файла для сервиса

```

victor: zsh — Konsole
[Новая вкладка] [Разделить окно]
[Копировать] [Вставить] [Поиск...]

anp 03 18:58:38 paradisePC systemd[1]: Stopped Single-User Mode Custom Script.
[+] sudo systemctl enable singleuser.service
[+] sudo systemctl status singleuser.service
○ singleuser.service - Single-User Mode Custom Script
  Loaded: loaded (/etc/systemd/system/singleuser.service; enabled; preset: disabled)
  Active: inactive (dead) since Thu 2025-04-03 18:58:38 MSK; 8min ago
  Duration: 2min 3.842s
  Invocation: dbb5e763aa9c4942aaaae7f3c4be72ed
  Main PID: 85363 (code=exited, status=0/SUCCESS)
  Mem peak: 1.4M
  CPU: 7ms

anp 03 18:56:34 paradisePC systemd[1]: Starting Single-User Mode Custom Script...
anp 03 18:56:34 paradisePC singleuser[85363]: [SYSTEMD] Running single-user tasks...
anp 03 18:56:34 paradisePC systemd[1]: Finished Single-User Mode Custom Script.
anp 03 18:58:38 paradisePC systemd[1]: Stopping Single-User Mode Custom Script...
anp 03 18:58:38 paradisePC singleuser[85448]: [SYSTEMD] Cleaning up single-user mode...
anp 03 18:58:38 paradisePC systemd[1]: singleuser.service: Deactivated successfully.
anp 03 18:58:38 paradisePC systemd[1]: Stopped Single-User Mode Custom Script.
[+] sudo systemctl start singleuser.service
[+] sudo systemctl status singleuser.service
● singleuser.service - Single-User Mode Custom Script
  Loaded: loaded (/etc/systemd/system/singleuser.service; enabled; preset: disabled)
  Active: active (exited) since Thu 2025-04-03 19:07:00 MSK; 1s ago
  Invocation: 7cb15fa5ac744ef892e60502b63cd116
  Process: 88361 ExecStart=/usr/local/bin/singleuser start (code=exited, status=0/SUCCESS)
  Main PID: 88361 (code=exited, status=0/SUCCESS)
  Mem peak: 1.6M
  CPU: 19ms

anp 03 19:07:00 paradisePC systemd[1]: Starting Single-User Mode Custom Script...
anp 03 19:07:00 paradisePC singleuser[88361]: [SYSTEMD] Running single-user tasks...
anp 03 19:07:00 paradisePC systemd[1]: Finished Single-User Mode Custom Script.

```

Рисунок 3 – запуск и проверка работы службы

Для проверки работы задания перешел в single mode

```
sudo systemctl rescue
```

Система перезагрузится и загрузится уже в однопользовательском режиме

Рисунок – проверка работы службы в single mode

```

etc > systemd > system > ≡ mounted-fs-logger.service
1  [Unit]
2  Description=Log mounted filesystems and time after boot
3  After=local-fs.target # Запускаем после монтирования всех локальных ФС
4  Requires=local-fs.target
5
6  [Service]
7  Type=oneshot
8  RemainAfterExit=yes
9  ExecStart=/bin/bash -c 'date >> /var/log/mounted-fs.log && mount >> /var/log/mounted-fs.log'
10
11 [Install]
12 WantedBy=multi-user.target
13 |

```

Рисунок – конфигурационный файл службы

2. В среде, содержащей систему systemd, описать новый тестовый системный юнит, который запускается после монтирования всех файловых систем и сохраняет список смонтированных систем и время в файл журнала

```

~$ sudo systemctl daemon-reload
~$ sudo systemctl enable mounted-fs-logger.service
Created symlink '/etc/systemd/system/multi-user.target.wants/mounted-fs-logger.service' → '/etc/systemd/system/mounted-fs-logger.service'.
~$ sudo systemctl start mounted-fs-logger.service
~$ sudo systemctl status mounted-fs-logger.service
● mounted-fs-logger.service - Log mounted filesystems and time after boot
   Loaded: loaded (/etc/systemd/system/mounted-fs-logger.service; enabled; preset: disabled)
   Active: active (exited) since Thu 2025-04-03 20:23:25 MSK; 5s ago
  Invocation: f454d55782f94f5e87c9f97cf6df42af
    Process: 92033 ExecStart=/bin/bash -c date >> /var/log/mounted-fs.log && mount >> /var/log/mounted-fs.log (code=exited, status=0/SUCCESS)
   Main PID: 92033 (code=exited, status=0/SUCCESS)
    Mem peak: 1.8M
      CPU: 14ms

apr 03 20:23:25 paradisePC systemd[1]: Starting Log mounted filesystems and time after boot...
apr 03 20:23:25 paradisePC systemd[1]: Finished Log mounted filesystems and time after boot.
lines 1-11/11 (END)

```

Рисунок — перезапуск менеджера служб, добавление службы в автозапуск и проверка работы службы



```

cat /var/log/mounted-fs.log
Чт 03 апр 2025 20:23:25 MSK
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sys on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
dev on /dev type devtmpfs (rw,nosuid,relatime,size=3964980k,nr_inodes=991245,mode=755,inode64)
run on /run type tmpfs (rw,nosuid,nodev,relatime,mode=755,inode64)
efivarfs on /sys/firmware/efi/efivars type efivarfs (rw,nosuid,nodev,noexec,relatime)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
/dev/nvme0n1p2 on / type ext4 (rw,relatime)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=40,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=6323)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,nosuid,nodev,relatime,pagesize=2M)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run/credentials/systemd-journald.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymfollow,size=1024k,nr_inodes=1024,mode=700,inode64,noswap)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /tmp type tmpfs (rw,noatime,inode64)
/var/lib/snapd/snaps/bare_5.snap on /var/lib/snapd/snap/bare/5 type squashfs (ro,nodev,relatime,errors=continue,thread=single,x-gdu.hide,x-gvfs-hide)
/var/lib/snapd/snaps/gnome-3-28-1804_198.snap on /var/lib/snapd/snap/gnome-3-28-1804/198 type squashfs (ro,nodev,relatime,errors=continue,threads=single,x-gdu.hide,x-gvfs-hide)
/var/lib/snapd/snaps/arduino_85.snap on /var/lib/snapd/snap/arduino/85 type squashfs (ro,nodev,relatime,errors=continue,threads=single,x-gdu.hide,x-gvfs-hide)

```

Рисунок — проверка работы

Как видим в лог файл успешно записалось время и все смонтированные разделы файловой системы

3. Создать тестовый скрипт и обеспечить его выполнения по расписанию каждую пятницу 2 недели каждого месяца в 01 часов 12 минут

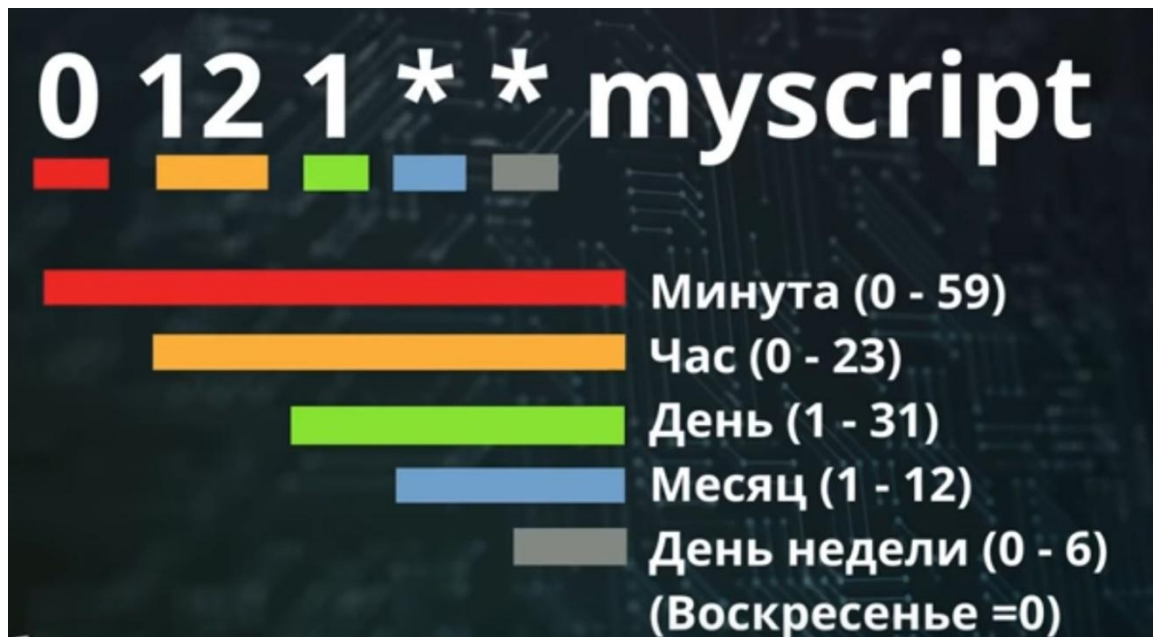
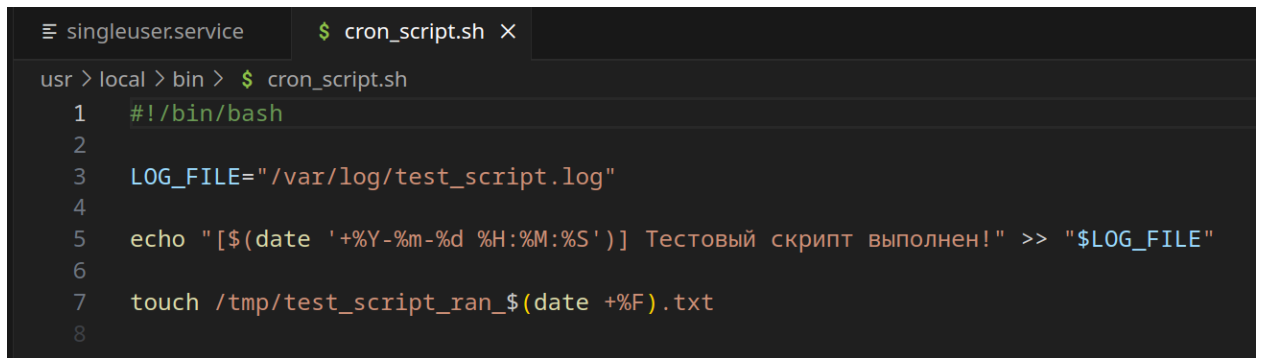
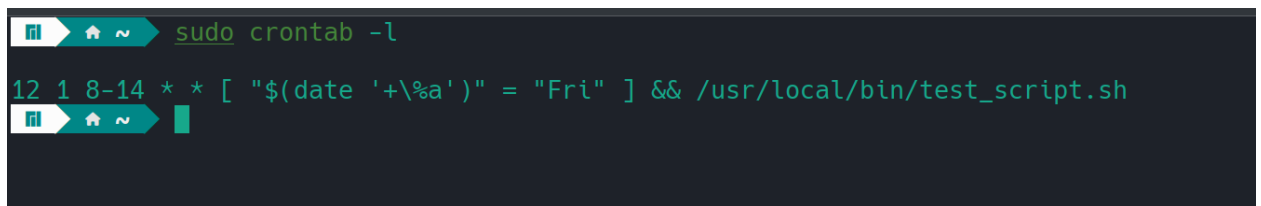


Рисунок 10 – синтаксис cron



```
singleuser.service  $ cron_script.sh X
usr > local > bin > $ cron_script.sh
1  #!/bin/bash
2
3  LOG_FILE="/var/log/test_script.log"
4
5  echo "[$(date '+%Y-%m-%d %H:%M:%S')] Тестовый скрипт выполнен!" >> "$LOG_FILE"
6
7  touch /tmp/test_script_ran_$(date +%F).txt
8
```

Рисунок 11 - Просмотр скрипта



```
12 1 8-14 * * [ "$(date '+\%a')" = "Fri" ] && /usr/local/bin/test_script.sh
```

Рисунок 12 — добавленное правило запуска скрипта в crontab