

# Лабораторная работа №14

Операционные системы

---

Луангсуваннавонг Сайпхачан

8 мая 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Луангсуваннавонг Сайпхачан
- Студент из группы НКАбд-01-24
- Российский университет дружбы народов
- <https://sayprachanh-lsvnv.github.io>

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

1. Написать командный файл, реализующий упрощённый механизм семафоров.

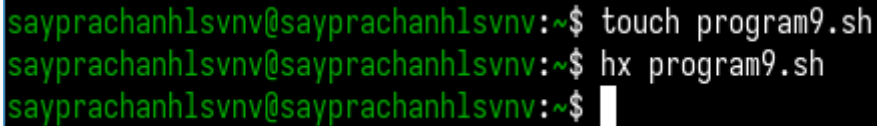
Командный файл должен в течение некоторого времени  $t_1$  дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени  $t_2 < t_1$ , также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом). Запустить командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой (`> /dev/tty#`, где `#` — номер терминала куда перенаправляется вывод), в котором также запущен этот файл, но не фоновом, а в привилегированном режиме. Доработать программу так, чтобы имелась возможность взаимодействия трёх и более процессов.

2. Реализовать команду `man` с помощью командного файла. Изучите содержимое каталога `/usr/share/man/man1`. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1`.
3. Используя встроенную переменную `$RANDOM`, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что `$RANDOM` выдаёт псевдослучайные числа в диапазоне от 0 до 32767.

## Выполнение лабораторной работы

---

Я создаю файл 'program9.sh' и открываю его для редактирования. (рис. 1)

A terminal window with a black background and green text. It shows three lines of commands and their prompts. The first line is 'touch program9.sh' with the prompt 'sayprachanh1svnv@sayprachanh1svnv:~\$'. The second line is 'hx program9.sh' with the same prompt. The third line shows the prompt 'sayprachanh1svnv@sayprachanh1svnv:~\$' followed by a white cursor block.

```
sayprachanh1svnv@sayprachanh1svnv:~$ touch program9.sh
sayprachanh1svnv@sayprachanh1svnv:~$ hx program9.sh
sayprachanh1svnv@sayprachanh1svnv:~$
```

Рис. 1: Создание файла



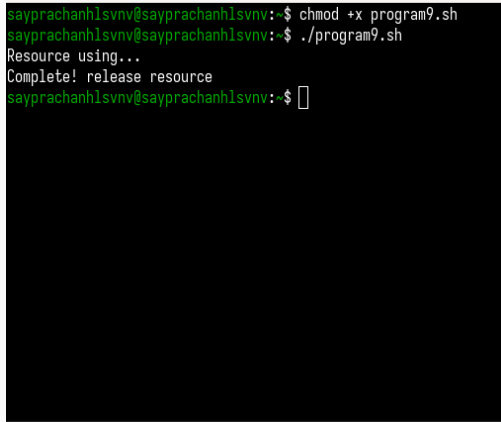
## Выполнение лабораторной работы

В файле 'program9.sh' я ввожу программу, которая реализует упрощённый механизм семафора с помощью команды 'flock'. Программа будет использовать разные времена для ожидания и использования ресурса. (рис. 2)

```
1  #!/bin/bash
2
3  t1=5
4  t2=7
5  lockfile="./lockfile"
6
7  while true;
8  do
9      exec {fn}>$lockfile
10
11     if flock -n $fn
12     then
13         echo "Resource using..."
14         sleep $t2
15         echo "Complete! release resource"
16         flock -u $fn
17
18     fi
```

## Выполнение лабораторной работы

Я запускаю программу. Программа во втором терминале ждёт, пока программа в первом терминале завершит свой процесс, выводит сообщение, а затем начинает и завершает свой процесс. (рис. 3)



```
sayprachanhlsnv@sayprachanhlsnv:~$ chmod +x program9.sh
sayprachanhlsnv@sayprachanhlsnv:~$ ./program9.sh
Resource using...
Complete! release resource
sayprachanhlsnv@sayprachanhlsnv:~$
```

Terminal - sayprachanhlsnv@sayprachanhlsnv:/home/sayprachanhlsnv


## Выполнение лабораторной работы

Далее я проверяю содержимое директории /usr/share/man/man1 для реализации следующей программы. (рис. 4)

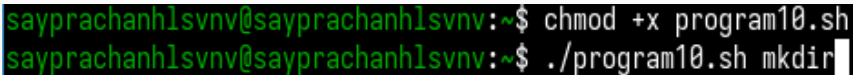
```
mknoo.1.gz      xzcat.1.gz
mkocp.1.gz      xzcmp.1.gz
mkofm.1.gz      xzdec.1.gz
mkpasswd.1.gz   xzdiff.1.gz
mkrfc2734.1.gz  xzegrep.1.gz
mksquashfs.1.gz xzfgrep.1.gz
mktemp.1.gz     xzgrep.1.gz
mktexfmt.1.gz   xzless.1.gz
mktexlsr.1.gz   xzmore.1.gz
mktexmf.1.gz    yes.1.gz
mktexpk.1.gz    ypdomainname.1.gz
mktextfm.1.gz   yt-dlp.1.gz
mlabel.1.gz     yum-changelog.1.gz
mmafm.1.gz      zcat.1.gz
mmcli.1.gz      zcmp.1.gz
mmd.1.gz        zdiff.1.gz
mmdblookup.1.gz zenity.1.gz
mmount.1.gz     zforce.1.gz
mmove.1.gz      zgrep.1.gz
mmpfb.1.gz      zip.1.gz
modulemd-validator.1.gz zipcloak.1.gz
moggsplit.1.gz  zipdetails.1.gz
mogrify.1.gz    zipgrep.1.gz
mokutil.1.gz    zipinfo.1.gz
montage.1.gz    zipnote.1.gz
more.1.gz       zipsplit.1.gz
mount.ddi.1.gz  zless.1.gz
mountpoint.1.gz zmore.1.gz
mpage.1.gz      znew.1.gz
```

## Выполнение лабораторной работы

Я создаю файл 'program10.sh' и ввожу программу, которая работает как команда man. Программа принимает имя команды в качестве аргумента командной строки, а затем с помощью команд 'less' и 'zcat' выводит информацию о команде. (рис. 5)

```
1   ./bin/bash
2
3  a=$1
4  if test -f /usr/share/man/man1/$a.1.gz
5      then zcat /usr/share/man/man1/$a.1.gz | less
6  else
7      echo "There is no such command"
8  fi
~
```

Я запускаю программу, ввожу имя команды, и она отображает справочную информацию по этой команде. (рис. 6 и рис. 7)



```
sayprachanhlsnv@sayprachanhlsnv:~$ chmod +x program10.sh  
sayprachanhlsnv@sayprachanhlsnv:~$ ./program10.sh mkdir
```

A terminal window with a black background and green text. The prompt is 'sayprachanhlsnv@sayprachanhlsnv:~\$'. The first command is 'chmod +x program10.sh'. The second command is './program10.sh mkdir' followed by a white cursor block.

Рис. 6: Запуск программы

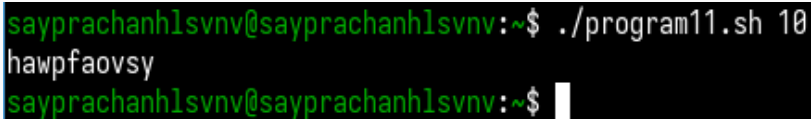
```
.\" DO NOT MODIFY THIS FILE! It was generated by help2man 1.48.5.
.TH MKDIR "1" "November 2024" "GNU coreutils 9.5" "User Commands"
.SH NAME
mkdir \- make directories
.SH SYNOPSIS
.B mkdir
[\fI\,OPTION\[/\fR]... \fI\,DIRECTORY\[/\fR...
.SH DESCRIPTION
.\" Add any additional description here
.PP
Create the DIRECTORY(ies), if they do not already exist.
.PP
Mandatory arguments to long options are mandatory for short options too.
.TP
\fB\[-m\fR, \fB\[-mode\fR=\fI\,MODE\[/\fR
```

## Выполнение лабораторной работы

Затем я создаю файл 'program11.sh' и ввожу программу, которая генерирует случайную последовательность букв латинского алфавита, используя встроенную переменную \$RANDOM. (рис. 8)

```
1 a=$1
2 letter=(a b c d e f g h i j k l m n o p q r s t u v w x y z)
3
4 for ((i=0; i<a; i++))
5 do
6     random=$((RANDOM % 26))
7
8     echo -n "${letter[$random]}"
9 done
10 echo
```

Я запускаю программу, ввожу число генерируемых букв в качестве аргумента, и она генерирует 10 случайных букв. (рис. 9)

A terminal window with a black background and green text. The prompt is 'sayprachanhlsvnnv@sayprachanhlsvnnv:~\$'. The user enters './program11.sh 10'. The program outputs 'hawpfaovsy'. The prompt returns to 'sayprachanhlsvnnv@sayprachanhlsvnnv:~\$' followed by a white cursor bar.

```
sayprachanhlsvnnv@sayprachanhlsvnnv:~$ ./program11.sh 10  
hawpfaovsy  
sayprachanhlsvnnv@sayprachanhlsvnnv:~$ █
```

Рис. 9: Запуск программы



Во время этой лабораторной работы я изучил основы программирования в оболочке UNIX. Научился писать более сложные пакетные файлы, используя логические структуры управления и циклы.

Спасибо за внимание

---