

Лабораторная работа №12

НКАбд-03-22

Шубнякова Дарья

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
5	Выводы	15
	Список литературы	16

Список иллюстраций

4.1	Программа	10
4.2	Реализация программы в терминале	10
4.3	Просмотр каталога с архивами руководств	11
4.4	Программа	12
4.5	Реализация программы	12
4.6	Реализация программы	13
4.7	Программа	14
4.8	Реализация программы	14

Список таблиц

1 Цель работы

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

2 Задание

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.

3 Теоретическое введение

Среди операционных систем особое место занимает Unix. Беспрецедентным является то, что ОС Unix может работать практически на всех выпускаемых платформах. UNIX - это стандарт де факто открытых и мобильных операционных систем. (поскольку название UNIX запатентовано компанией AT&T - различные юниксы называются различно: SCO UNIX, BSDI, Solaris, Linux, DG/UX, AIX и т.д.).

4 Выполнение лабораторной работы

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.

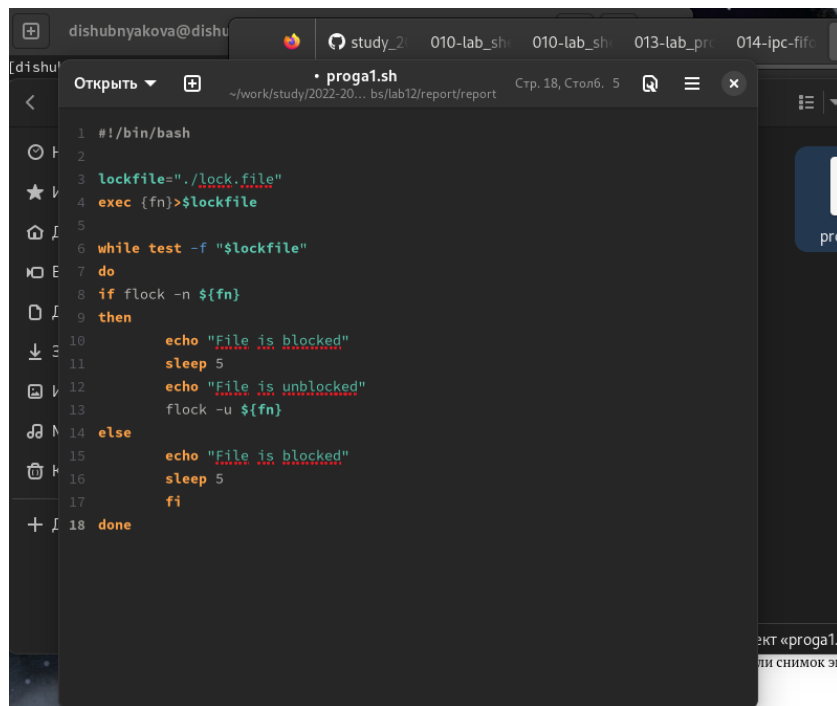


Рис. 4.1: Программа

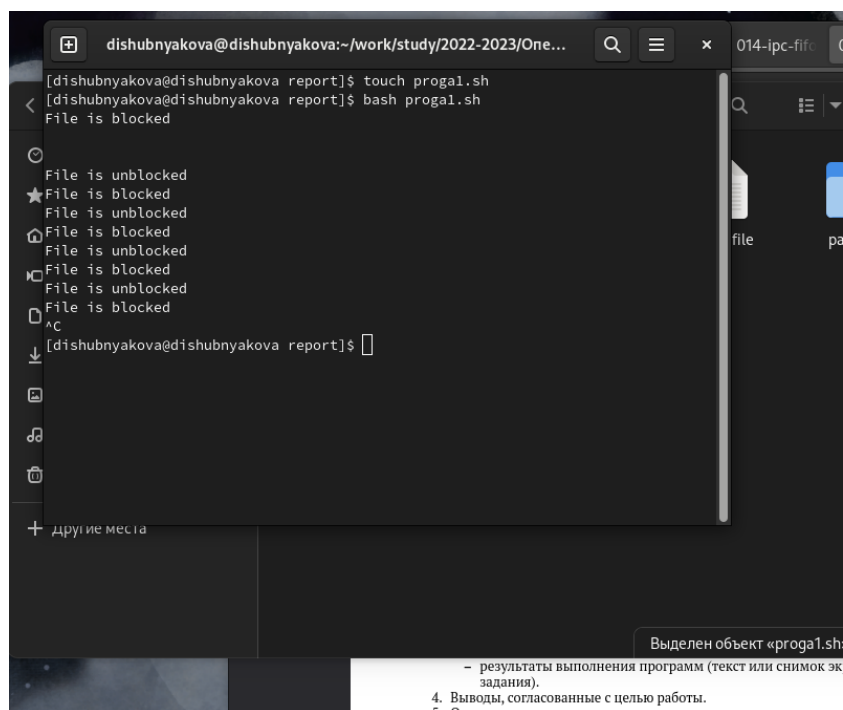


Рис. 4.2: Реализация программы в терминале

Программа команды `man`. 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`.

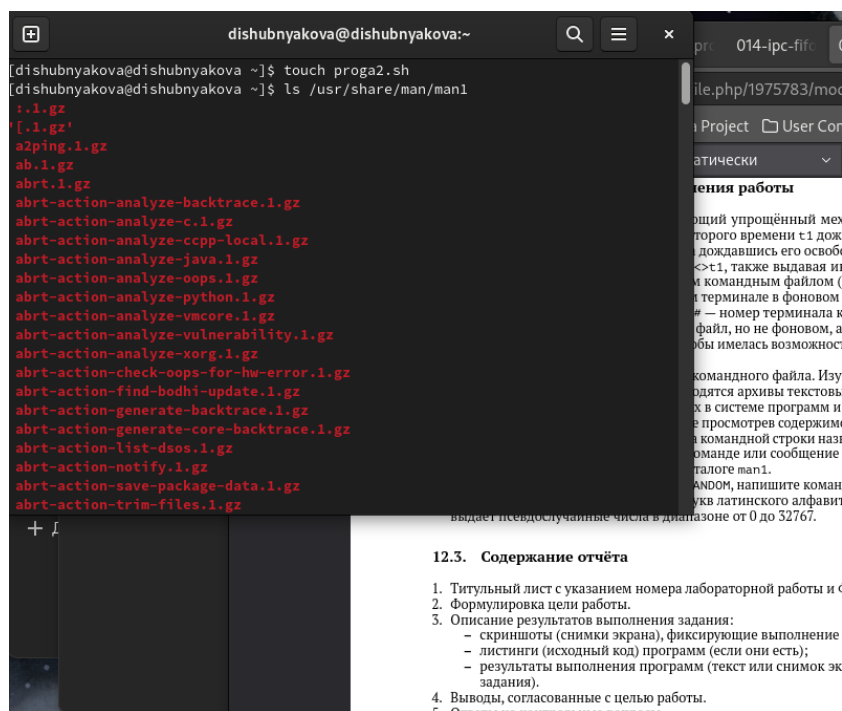


Рис. 4.3: Просмотр каталога с архивами руководств

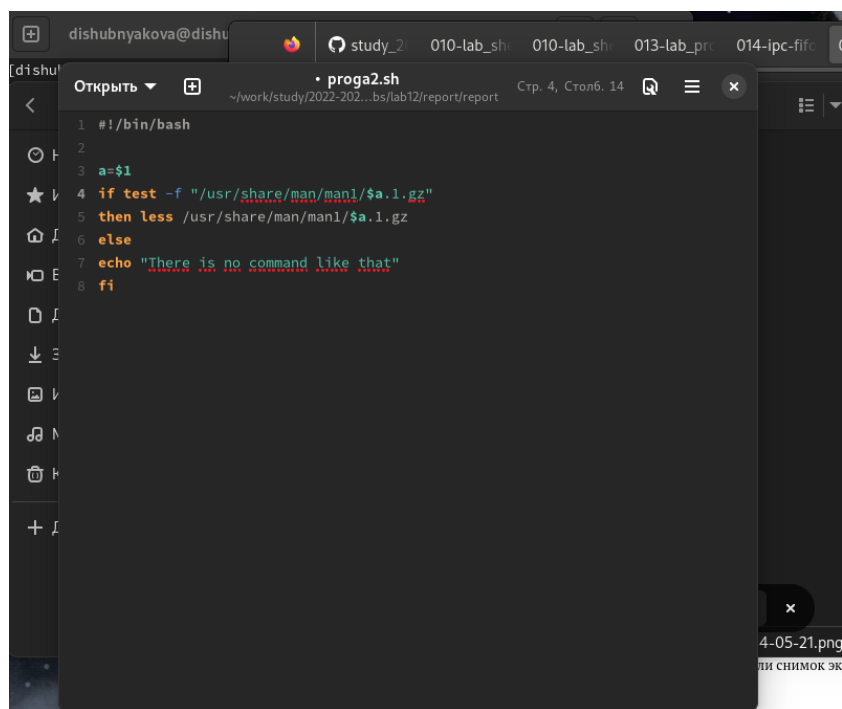


Рис. 4.4: Программа

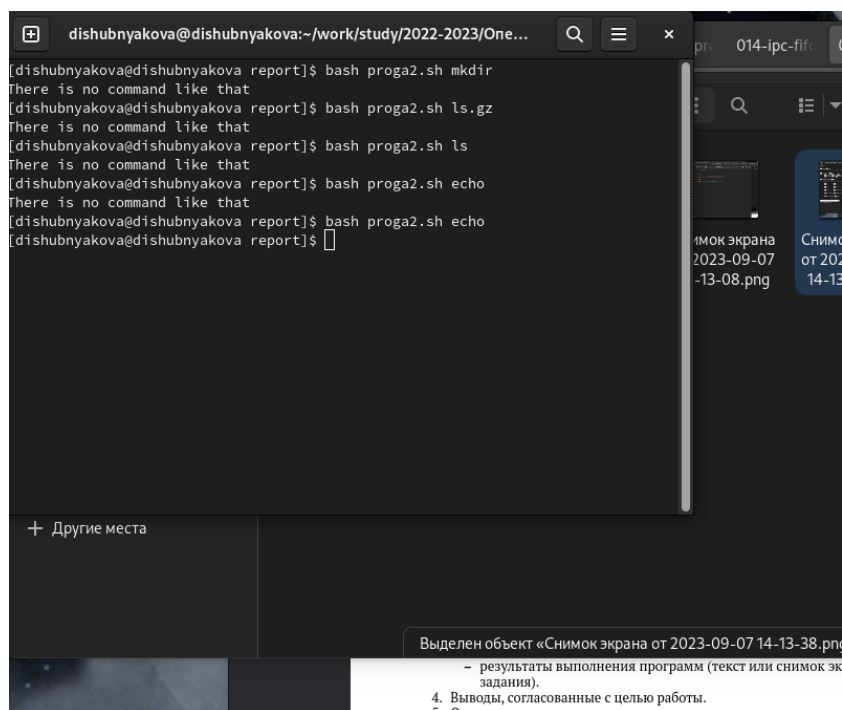


Рис. 4.5: Реализация программы

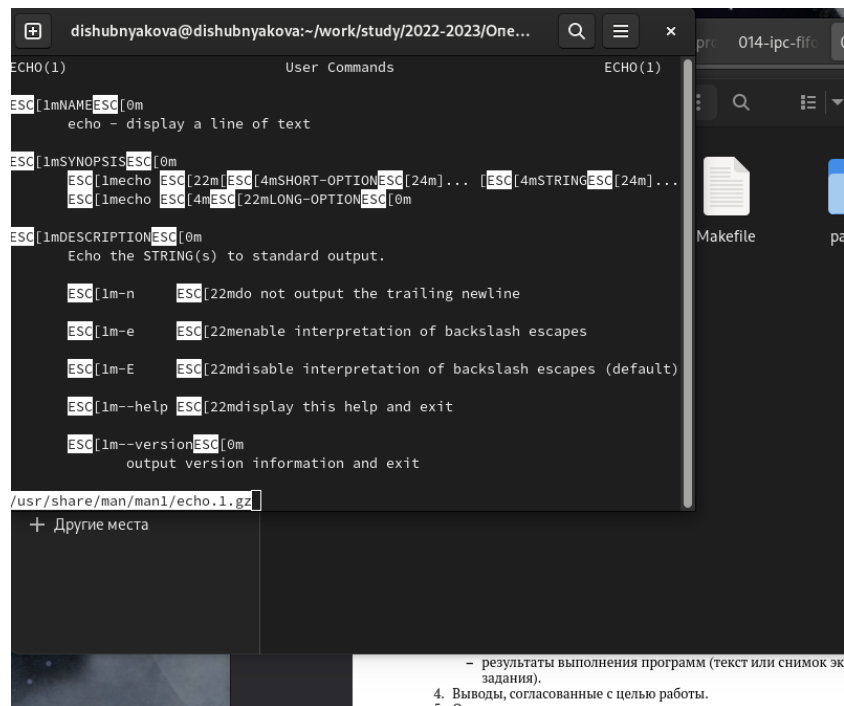
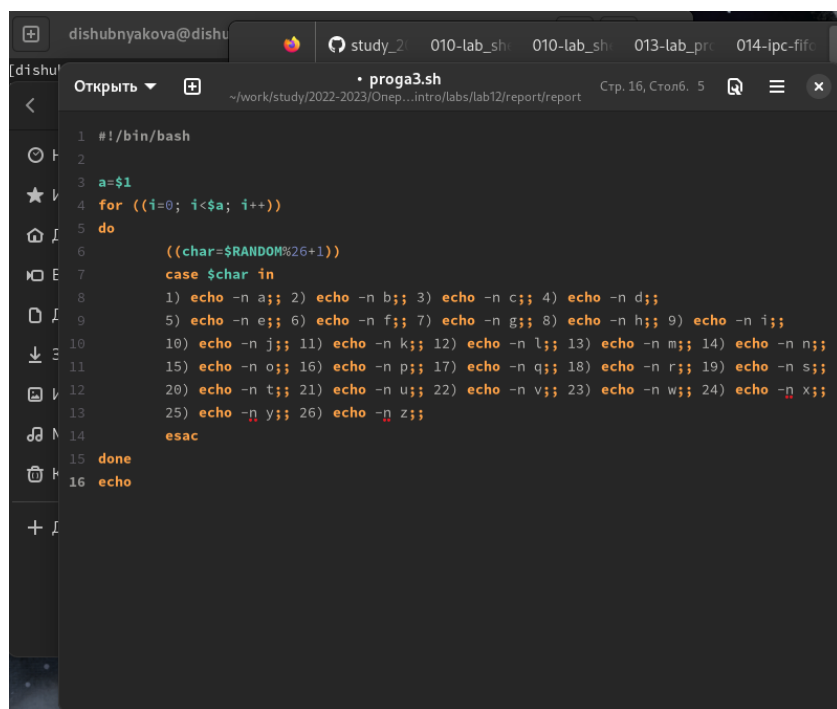


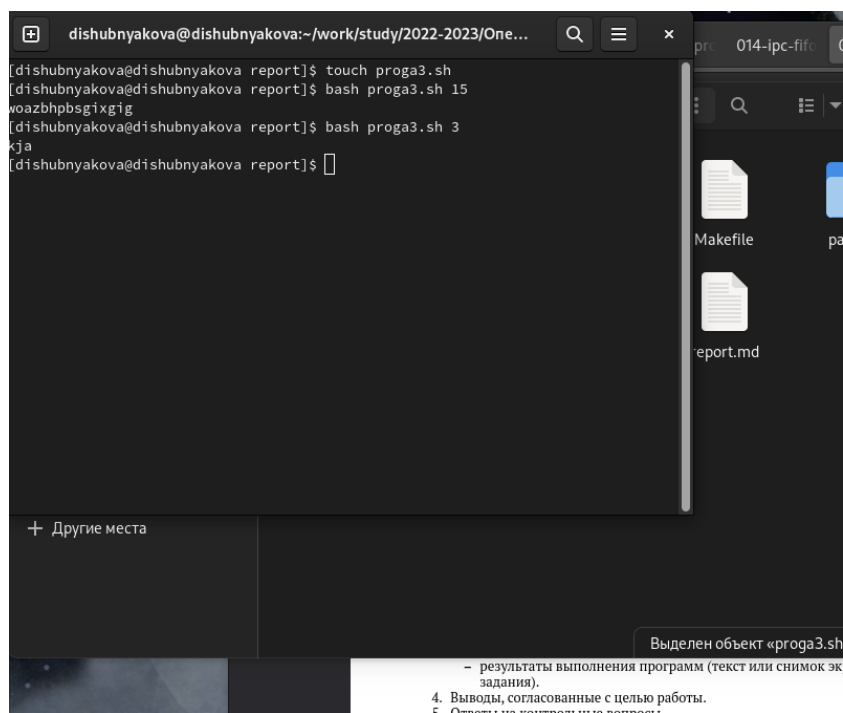
Рис. 4.6: Реализация программы

Командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что \$RANDOM выдаёт псевдослучайные числа в диапазоне от 0 до 32767.



```
1 #!/bin/bash
2
3 a=$1
4 for ((i=0; i<$a; i++))
5 do
6     ((char=$RANDOM%26+1))
7     case $char in
8         1) echo -n a;; 2) echo -n b;; 3) echo -n c;; 4) echo -n d;;
9         5) echo -n e;; 6) echo -n f;; 7) echo -n g;; 8) echo -n h;; 9) echo -n i;;
10        10) echo -n j;; 11) echo -n k;; 12) echo -n l;; 13) echo -n m;; 14) echo -n n;;
11        15) echo -n o;; 16) echo -n p;; 17) echo -n q;; 18) echo -n r;; 19) echo -n s;;
12        20) echo -n t;; 21) echo -n u;; 22) echo -n v;; 23) echo -n w;; 24) echo -n x;;
13        25) echo -n y;; 26) echo -n z;;
14    esac
15 done
16 echo
```

Рис. 4.7: Программа



```
[dishubnyakova@dishubnyakova report]$ touch proga3.sh
[dishubnyakova@dishubnyakova report]$ bash proga3.sh 15
woazbhpbgixgig
[dishubnyakova@dishubnyakova report]$ bash proga3.sh 3
kja
[dishubnyakova@dishubnyakova report]$
```

Рис. 4.8: Реализация программы

5 Выводы

Составили три сложные программы в оболочке ОС UNIX.

Список литературы