

Отчет по лабораторной работе № 2 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Ефименко Кирилл Игоревич, № по списку 6

Контакты



Работа выполнена: «12» сентября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « _____ » _____ 20__ г., итоговая оценка _____

Подпись преподавателя

1. Тема: Операционная среда ОС UNIX

2. Цель работы: взять цель работы из текста лр

3. Задание (вариант № номер варианта, если есть): Приобретение основных навыков работы в ОС UNIX

4. Оборудование (студента):

Процессор AMD ryzen 7 5800U 8x 3.9GH с ОП 16384 Мб, НМД 512Гб. Монитор 1920x1080

5. Программное обеспечение (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *18.10 cosmic*
интерпретатор команд: *bash* версия *4.4.19*.

Система программирования -- версия --, редактор текстов *emacs* версия *25.2.2*

Утилиты операционной системы --

Прикладные системы и программы --

Местонахождение и имена файлов программ и данных на домашнем компьютере --

6. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

```
#!/bin/bash
echo "В какой директории вы хотите удалить пустые файлы? Y - выбрать текущую, N -
ввести вручную"
choice=""
read choice
if [ $choice = "Y" ]
then
mydir=$(pwd)
emptyFiles=$(find $mydir -empty)
    if rm $emptyFiles
    then
        echo "Пустые файлы были удалены, скрипт завершен"
    else
        echo "Пустые файлы не найдены"
    fi
elif [ $choice = "N" ]
then
mydir=""
read mydir
emptyFiles=$(find $mydir -empty)
    if rm $emptyFiles
    then
        echo "Пустые файлы были удалены, скрипт завершен"
    else
        echo "Пустые файлы не найдены"
    fi
else
echo "Вы ввели неправильные данные, перезапустите скрипт"
fi
```

7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Алгоритм:

- 1) Узнать папку, в которой нужно удалить все пустые файлы
- 2) Создать переменную, хранящую все пустые файлы
- 3) Передать переменную с файлами оператору rm
- 4) Вывести результат

8. **Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

```
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works  program.txt  test
newuser@LAPTOP-HO30I6QD:~$ rm -r Lab_works
rm: remove write-protected regular file
'Lab_works/.git/objects/d7/7296efab9f5a02c085971e9b7a48c204870516'?
rm: cannot remove 'Lab_works/.git/objects/d7': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/5f/2be496bd06d0a35469160fe61152f850b7f038'? y
```

```
rm: remove write-protected regular file
'Lab_works/.git/objects/f7/bb12c739eb9072a9b148d16f4445a3e9d98545'?
rm: cannot remove 'Lab_works/.git/objects/f7': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/8b/137891791fe96927ad78e64b0aad7bdc'? y
rm: remove write-protected regular file
'Lab_works/.git/objects/fe/df874a701f13d8c17134f3d92adcf20bc1800e'?
rm: cannot remove 'Lab_works/.git/objects/fe': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/42/0b445c355759e2a47e35fce21543b58b6ed975'?
rm: cannot remove 'Lab_works/.git/objects/42': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/94/275eb506fa0e34f43eded1ea1ca6b87ec213ca'?
rm: cannot remove 'Lab_works/.git/objects/94': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/e6/9de29bb2d1d6434b8b29ae775ad8c2e48c5391'?
rm: cannot remove 'Lab_works/.git/objects/e6': Directory not empty
newuser@LAPTOP-HO30I6QD:~$
newuser@LAPTOP-HO30I6QD:~$
newuser@LAPTOP-HO30I6QD:~$
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works  program.txt  test
newuser@LAPTOP-HO30I6QD:~$ rm -d test
rm: cannot remove 'test': Directory not empty
newuser@LAPTOP-HO30I6QD:~$
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works  program.txt  test
newuser@LAPTOP-HO30I6QD:~$ rm -r test
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works  program.txt
newuser@LAPTOP-HO30I6QD:~$ rm program.txt
newuser@LAPTOP-HO30I6QD:~$ cd Lab_works
newuser@LAPTOP-HO30I6QD:~/Lab_works$ ls
newuser@LAPTOP-HO30I6QD:~/Lab_works$
newuser@LAPTOP-HO30I6QD:~/Lab_works$ rm -d Lab_works
rm: cannot remove 'Lab_works': No such file or directory
newuser@LAPTOP-HO30I6QD:~/Lab_works$ rm -r Lab_works
rm: cannot remove 'Lab_works': No such file or directory
newuser@LAPTOP-HO30I6QD:~/Lab_works$ ls
newuser@LAPTOP-HO30I6QD:~/Lab_works$ cd ..
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works
newuser@LAPTOP-HO30I6QD:~$ rm -d Lab_works
rm: cannot remove 'Lab_works': Directory not empty
newuser@LAPTOP-HO30I6QD:~$ rm -r Lab_works
rm: remove write-protected regular file
'Lab_works/.git/objects/d7/7296efab9f5a02c085971e9b7a48c204870516'?
rm: cannot remove 'Lab_works/.git/objects/d7': Directory not empty
rm: remove write-protected regular file
'Lab_works/.git/objects/f7/bb12c739eb9072a9b148d16f4445a3e9d98545'? y
rm: remove write-protected regular file
'Lab_works/.git/objects/fe/df874a701f13d8c17134f3d92adcf20bc1800e'? y
rm: remove write-protected regular file
'Lab_works/.git/objects/42/0b445c355759e2a47e35fce21543b58b6ed975'? y
rm: remove write-protected regular file
'Lab_works/.git/objects/94/275eb506fa0e34f43eded1ea1ca6b87ec213ca'? y
```

```
rm: remove write-protected regular file
'Lab_works/.git/objects/e6/9de29bb2d1d6434b8b29ae775ad8c2e48c5391'? y
newuser@LAPTOP-HO30I6QD:~$ y
yy: command not found
newuser@LAPTOP-HO30I6QD:~$ yls
```

Command 'yls' not found, did you mean:

```
command 'hls' from deb hfsutils (3.2.6-14)
command 'jls' from deb sleuthkit (4.6.7-1build1)
command 'ils' from deb sleuthkit (4.6.7-1build1)
command 'bls' from deb bacula-sd (9.4.2-2ubuntu5)
command 'tyls' from deb terminology (1.6.0-2)
command 'yes' from deb coreutils (8.30-3ubuntu2)
command 'ls' from deb coreutils (8.30-3ubuntu2)
command 'ols' from deb speech-tools (1:2.5.0-8build1)
command 'fls' from deb sleuthkit (4.6.7-1build1)
command 'als' from deb atool (0.39.0-10)
```

Try: sudo apt install <deb name>

```
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works
newuser@LAPTOP-HO30I6QD:~$ mkdir pool1
newuser@LAPTOP-HO30I6QD:~$ rm -d pool1
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works
newuser@LAPTOP-HO30I6QD:~$ mkdir pool{1..5}
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works pool1 pool2 pool3 pool4 pool5
newuser@LAPTOP-HO30I6QD:~$ cd pool1
newuser@LAPTOP-HO30I6QD:~/pool1$ touch file{1..3}.txt
newuser@LAPTOP-HO30I6QD:~/pool1$ ls
file1.txt file2.txt file3.txt
newuser@LAPTOP-HO30I6QD:~/pool1$ cd ..
newuser@LAPTOP-HO30I6QD:~$ cd pool2
newuser@LAPTOP-HO30I6QD:~/pool2$ touch cock{12..18}.py
newuser@LAPTOP-HO30I6QD:~/pool2$ ls
cock12.py cock13.py cock14.py cock15.py cock16.py cock17.py cock18.py
newuser@LAPTOP-HO30I6QD:~/pool2$ rm cock{12..18}.py
newuser@LAPTOP-HO30I6QD:~/pool2$ ls
newuser@LAPTOP-HO30I6QD:~/pool2$ touch cock{12..18}.py
newuser@LAPTOP-HO30I6QD:~/pool2$ ls
cock12.py cock13.py cock14.py cock15.py cock16.py cock17.py cock18.py
newuser@LAPTOP-HO30I6QD:~/pool2$ cd ..
newuser@LAPTOP-HO30I6QD:~$ pwd
/home/newuser
newuser@LAPTOP-HO30I6QD:~$ cd pool1
newuser@LAPTOP-HO30I6QD:~/pool1$ nano file1.txt
newuser@LAPTOP-HO30I6QD:~/pool1$ cat file1.txt
qwerty
1234
kikopq
a'gnsa
japegjp
newuser@LAPTOP-HO30I6QD:~/pool1$ cd ..
```

```
newuser@LAPTOP-HO30I6QD:~$ cd pool2
newuser@LAPTOP-HO30I6QD:~/pool2$ nano cock12.py
newuser@LAPTOP-HO30I6QD:~/pool2$ python3 cock12.py
1
2
3
4
5
6
7
8
9
newuser@LAPTOP-HO30I6QD:~/pool2$ cat cock12.py
a = 1
b = 10
for i in range(a, b):
    print(i);
newuser@LAPTOP-HO30I6QD:~/pool2$ cd ..
newuser@LAPTOP-HO30I6QD:~$ find -name cock12.py
./pool2/cock12.py
newuser@LAPTOP-HO30I6QD:~$ find -not -name *.txt
.
./octave_hist
./pool5
./gnuplot_history
./sudo_as_admin_successful
./cache
./cache/fontconfig
./cache/fontconfig/1ff91a00-c753-40fb-96b4-d63585744671-le64.cache-7
./cache/fontconfig/CACHEDIR.TAG
./Lab_works
./Lab_works/.git
./Lab_works/.git/objects
./Lab_works/.git/objects/d7
./Lab_works/.git/objects/d7/7296efab9f5a02c085971e9b7a48c204870516
./bash_history
./pool1
./pool2
./pool2/cock16.py
./pool2/cock14.py
./pool2/cock13.py
./pool2/cock15.py
./pool2/cock18.py
./pool2/cock12.py
./pool2/cock17.py
./config
./config/octave
./config/octave/qt-settings
./pool3
./profile
./bash_logout
./pool4
./viminfo
./bashrc
./local
./local/share
./local/share/nano
```

```

./dbus
./dbus/session-bus
./dbus/session-bus/e01e2e849eea562030d87586631b7cb9-0
newuser@LAPTOP-HO30I6QD:~$ find -namr *.py
find: unknown predicate `-namr'
newuser@LAPTOP-HO30I6QD:~$ find -name *.py
./pool2/cock16.py
./pool2/cock14.py
./pool2/cock13.py
./pool2/cock15.py
./pool2/cock18.py
./pool2/cock12.py
./pool2/cock17.py
newuser@LAPTOP-HO30I6QD:~$ mv file1.txt file228.go
mv: cannot stat 'file1.txt': No such file or directory
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works pool1 pool2 pool3 pool4 pool5
newuser@LAPTOP-HO30I6QD:~$ cd pool1
newuser@LAPTOP-HO30I6QD:~/pool1$ ls
file1.txt file2.txt file3.txt
newuser@LAPTOP-HO30I6QD:~/pool1$ mv file1.txt file228.go
newuser@LAPTOP-HO30I6QD:~/pool1$ ls
file2.txt file228.go file3.txt
newuser@LAPTOP-HO30I6QD:~/pool1$ cd ..\
> >
newuser@LAPTOP-HO30I6QD:~$ cd ..
newuser@LAPTOP-HO30I6QD:/home$ ls
newuser
newuser@LAPTOP-HO30I6QD:/home$ cd newuser
newuser@LAPTOP-HO30I6QD:~$ grep -i print
newuser@LAPTOP-HO30I6QD:~$ cd pool2
newuser@LAPTOP-HO30I6QD:~/pool2$ grep -i print
1
newuser@LAPTOP-HO30I6QD:~/pool2$ grep a
newuser@LAPTOP-HO30I6QD:~/pool2$ grep --help
Usage: grep [OPTION]... PATTERNS [FILE]...
Search for PATTERNS in each FILE.
Example: grep -i 'hello world' menu.h main.c
PATTERNS can contain multiple patterns separated by newlines.

```

Pattern selection and interpretation:

-E, --extended-regexp	PATTERNS are extended regular expressions
-F, --fixed-strings	PATTERNS are strings
-G, --basic-regexp	PATTERNS are basic regular expressions
-P, --perl-regexp	PATTERNS are Perl regular expressions
-e, --regexp=PATTERNS	use PATTERNS for matching
-f, --file=FILE	take PATTERNS from FILE
-i, --ignore-case	ignore case distinctions in patterns and data
--no-ignore-case	do not ignore case distinctions (default)
-w, --word-regexp	match only whole words
-x, --line-regexp	match only whole lines
-z, --null-data	a data line ends in 0 byte, not newline

Miscellaneous:

-s, --no-messages	suppress error messages
-v, --invert-match	select non-matching lines
-V, --version	display version information and exit

--help display this help text and exit

Output control:

-m, --max-count=NUM stop after NUM selected lines
-b, --byte-offset print the byte offset with output lines
-n, --line-number print line number with output lines
 --line-buffered flush output on every line
-H, --with-filename print file name with output lines
-h, --no-filename suppress the file name prefix on output
 --label=LABEL use LABEL as the standard input file name prefix
-o, --only-matching show only nonempty parts of lines that match
-q, --quiet, --silent suppress all normal output
 --binary-files=TYPE assume that binary files are TYPE;
 TYPE is 'binary', 'text', or 'without-match'
-a, --text equivalent to --binary-files=text
-I equivalent to --binary-files=without-match
-d, --directories=ACTION how to handle directories;
 ACTION is 'read', 'recurse', or 'skip'
-D, --devices=ACTION how to handle devices, FIFOs and sockets;
 ACTION is 'read' or 'skip'
-r, --recursive like --directories=recurse
-R, --dereference-recursive likewise, but follow all symlinks
 --include=GLOB search only files that match GLOB (a file pattern)
 --exclude=GLOB skip files that match GLOB
 --exclude-from=FILE skip files that match any file pattern from FILE
 --exclude-dir=GLOB skip directories that match GLOB
-L, --files-without-match print only names of FILES with no selected lines
-l, --files-with-matches print only names of FILES with selected lines
-c, --count print only a count of selected lines per FILE
-T, --initial-tab make tabs line up (if needed)
-Z, --null print 0 byte after FILE name

Context control:

-B, --before-context=NUM print NUM lines of leading context
-A, --after-context=NUM print NUM lines of trailing context
-C, --context=NUM print NUM lines of output context
-NUM same as --context=NUM
 --color[=WHEN],
 --colour[=WHEN] use markers to highlight the matching strings;
 WHEN is 'always', 'never', or 'auto'
-U, --binary do not strip CR characters at EOL (MSDOS/Windows)

When FILE is '-', read standard input. With no FILE, read '.' if recursive, '-' otherwise. With fewer than two FILES, assume -h. Exit status is 0 if any line (or file if -L) is selected, 1 otherwise; if any error occurs and -q is not given, the exit status is 2.

Report bugs to: bug-grep@gnu.org

GNU grep home page: <http://www.gnu.org/software/grep/>

General help using GNU software: <https://www.gnu.org/gethelp/>

```
newuser@LAPTOP-HO30I6QD:~/pool2$ grep . print
```

```
grep: print: No such file or directory
```

```
newuser@LAPTOP-HO30I6QD:~/pool2$ grep -vr print
```

```
cock12.py:a = 1
```

```
cock12.py:b = 10
```

```
cock12.py:for i in range(a, b):
```

```
newuser@LAPTOP-HO30I6QD:~/pool2$ cd ..
```

```
newuser@LAPTOP-HO30I6QD:~$ cat > file2.txt
qwertydohasuhgsdah;ghso[ahg
newuser@LAPTOP-HO30I6QD:~$ ls
Lab_works file2.txt pool1 pool2 pool3 pool4 pool5
newuser@LAPTOP-HO30I6QD:~$ cat file2.txt
qwertydohasuhgsdah;ghso[ahg
newuser@LAPTOP-HO30I6QD:~$...
```

9. Дневник отладки должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	лаб	12.09.2020	17:20	Создал не тот файл	Удалил и создал заново	Мне грустно

10. Замечания автора по существу работы
Замечаний нет

11. Выводы

Не знаю, насколько сильно мне понадобятся полученные знания, но было интересно изучить основы ОС UNIX. Думаю, я научился использовать элементарные команды в терминале UNIX. Много времени потратил на написание BASH скрипта, но потраченного времени не жалею, т. к. было интересно изучать синтаксис нового для меня языка.

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента
