

Лабораторная работа 13

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Чигладзе М.В.

29 мая 2003

Российский университет дружбы народов, Москва, Россия

Информация

- Чигладзе Майя Владиславовна
- студентка РУДН направления Прикладная информатика
- заместитель ОСК профсоюза РУДН
- волонтер университета и Москвы
- [1132239399@pfur.ru]
- <https://github.com/LaMeru>



Вводная часть

- Актуальна для студентов, изучающих операционные системы и программирование в контексте Unix-систем.

- Объект исследования: Командный процессор операционной системы UNIX, его синтаксис и возможности.
- Предмет исследования: Написание и выполнение командных файлов на языке UNIX Shell.

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

Задачи работы:

- Изучение возможностей и синтаксиса командного процессора UNIX
- Освоение основных операторов и конструкций языка UNIX Shell
- Изучение методов обработки аргументов и опций
- Освоение операторов ветвления и циклов
- Изучение возможностей управления ошибками и выхода из программ
- Разработка и выполнение лабораторных заданий.

Методы исследования включают изучение документации и примеров использования командного процессора, написание и тестирование командных файлов, анализ результатов выполнения команд.

Выполнение лабораторной работы

Задание 1.

Задача: Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: – `-iinputfile` — прочитать данные из указанного файла; – `-ooutputfile` — вывести данные в указанный файл; – `-rшаблон` — указать шаблон для поиска; – `-C` — различать большие и малые буквы; – `-n` — выдавать номера строк.

```
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# nano first.md
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# grep -i -o -p -C -n 'Maiaa' first.md first_output.md
grep: неверный ключ - «р»
Использование: grep [ПАРАМЕТР]... ШАБЛОНЫ [ФАЙЛ]...
Запустите «grep --help» для получения более подробного описания.
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# grep -i -o -p -C -n 'Maiaa' first.md first_output.md
grep: неверный ключ - «р»
Использование: grep [ПАРАМЕТР]... ШАБЛОНЫ [ФАЙЛ]...
Запустите «grep --help» для получения более подробного описания.
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# grep -i -o -C -n 'Maiaa' first.md first_output.md
grep: -n: неверный аргумент длины контекста
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# grep -i -o -p 'Maiaa' first.md first_output.md
grep: неверный ключ - «р»
Использование: grep [ПАРАМЕТР]... ШАБЛОНЫ [ФАЙЛ]...
Запустите «grep --help» для получения более подробного описания.
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# grep -i -o -p "Maiaa" first.md first_output.md
grep: неверный ключ - «р»
Использование: grep [ПАРАМЕТР]... ШАБЛОНЫ [ФАЙЛ]...
Запустите «grep --help» для получения более подробного описания.
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ls
first.md
```

Рис. 1: Код

Задание 2.

Задача: Написать на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию в о коде завершения в оболочку. Команд- ный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

```
GNU nano 7.2
#!/bin/bash

# Путь до файла с кодом на языке C
FILE_PATH="/home/mvchigladze/Загрузки/second.c"

# Вызов компилятора и запуск программы
gcc $FILE_PATH -o program && ./program

# Проверка кода завершения программы
if [ "$?" -eq 0 ]; then
    echo "Программа завершилась успешно."
else
    echo "Произошла ошибка."
fi
```

Рис. 2: Код

```
GNU nano 7.2
#include <stdio.h>
int main() {
int number;
printf("Введите число: ");
scanf("%d", &number);
if (number > 0) {
    printf("Число больше нуля.\n");
} else if (number < 0) {
    printf("Число меньше нуля.\n");
} else {
    printf("Число равно нулю.\n");
}

return 0;
}
```

```
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ./second.sh
Введите число: 14
Число больше нуля.
Программа завершилась успешно.
```

Рис. 4: Результат

Задание 3.

Задача: Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до ∞ (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).


```
GNU nano 7.2
#!/bin/bash

function create_files() {
    local N="$1"
    for ((i=1; i<=N; i++)); do
        touch "$i.tmp"
    done
}

function delete_files() {
    local files_to_delete=(*.tmp)
    rm -f "${files_to_delete[@]}"
}

if [[ $# -eq 0 ]]; then
    echo "Необходимо указать количество файлов для создания."
    exit 1
fi

create_files "$1"
delete_files
```

```
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# chmod +x third.sh
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ls
first.md  program  second.c  second.sh  third.sh
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ./third.sh
Необходимо указать количество файлов для создания.
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ./third.sh 4
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ls
first.md  program  second.c  second.sh  third.sh
```

Рис. 6: Результат

Задание 4.

Задача: Написать командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`).

```
GNU nano 7.2
# Командный файл для выполнения операций с использованием grep, tar и find

# Аргументы командной строки: директория, в которой нужно найти файлы, и имя архива
if [ -z "$1" ] || [ -z "$2" ]; then
    echo "Использование: `basename $0` директория архив"
    exit 1
fi

# Поиск файлов, измененных менее недели назад
find "$1" -newermt '7 days ago' -print0 |

# Передача списка файлов в tar для создания архива
xargs -0 tar -cvzf "$2".tar.gz

# Удаление исходных файлов
find "$1" -newermt '7 days ago' -exec rm -f {} +
```

Рис. 7: Код

```
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ./thourth.sh /home/mvchigladze/Рабочий стол/study_2022
Labs/ LICENSE
root@mvchigladze:/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13# ./thourth.sh /home/mvchigladze/Рабочий стол/study_2022
tar: Удаляется начальный '/' из имен объектов
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/first.md
tar: Удаляются начальные '/' из целей жестких ссылок
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/second.c
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/second.sh
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/program
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/third.sh
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/thourth.sh
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/first.md
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/second.c
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/second.sh
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/program
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/third.sh
/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13/thourth.sh
гм: невозможно удалить '/home/mvchigladze/Рабочий стол/study_2022-2023_os-intro/Labs/lab13': Это каталог
```

Рис. 8: Результат

Результаты

В результате данной работы я изучила основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.