МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

## ***Факультет информационных технологий и робототехники***

Кафедра программного обеспечения информационных систем и технологий

**Отчет по лабораторной работе № 3**

по дисциплине:” Системное программирование ”

на тему: ”Командный язык и скрипты SHELL***”***

Выполнил**:** студент группы 10701321 Кругликовский М.А.

Принял**:** ст.пр. Станкевич C.П.

Минск 2023

# **Лабораторная работа № 3**

**Цель работы**

Закрепить на практике основы работы со сценариями командной оболочки и перенаправление ввода/вывода.

**Задание**

**Задание 1**

1. Написать скрипт, который записывает содержимое одного файла в

другой и при этом оба файла существуют до операции. Дать два разных

варианта выполнения.

2. Найти все файлы, начинающиеся на слово script и записать их имена в

файл list.txt.

3. Написать скрипт, который проверяет, содержится ли скриптовый файл в

директории, если да, то выполнить его.

4. Написать скрипт, подсчитывающий сумму от 1 до 10.

5. А какие командные интерпретаторы установлены в вашей системе?

Напишите скрипт поиска местонахождения командных интерпретаторов вашей системы. Также определите какой интерпретатор используется сейчас в вашей системе?

**Задание 2**

6. Используйте текст первого примера, заменить некоторые фамилии на

фамилии членов вашей команды (например, Nutrina заменить на

Stankewich).

7. Отсортируйте данные файла так, чтобы он не поменял свое собственное имя. (Решение. Нужно предварительно создать промежуточный отсортированный файл t11.txt. Затем выполнить нужные команды).

8. Подсчитайте число одинаковых слов в обоих файлах. Для подсчета

числа слов в файле используйте команду: wc -w file1.

9. Написать скрипт, который создает отсортированный файл, содержащий

слова из двух файлов, исключая их общую часть одинаковых слов.

10.Вывести фамилию самого молодого человека (в файле basa.txt – это второй столбец).

11.Вывести зарплату самого молодого человека (зарплата – третий столбец).

12.Вывести отсортированный список имен файлов, начальная часть имени

есть script.

13.Написать скрипт, который в каждой строке файла оставляет только

буквенные символы, а остальные символы выбрасывает.

14.Вывести упорядоченный список имен файлов, в именах которых содержится символ подчеркивания, например, script\_sort.

**Задание 3**

15.Напишите скрипт запуска другого скрипта.

16.Расположите скрипт в одной из стантадртных директорий системы, и

запустите его из домашней директории пользователя.

**Результаты выполнения**

**Задание 1**

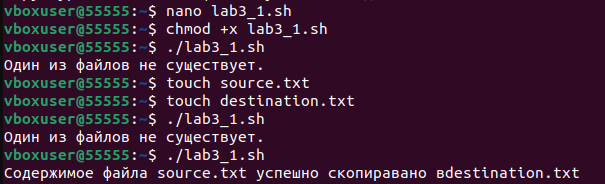
****

Рисунок 1. Скрипт для копирования содержимого одного файла в другой

if [ -e source.txt ] && [ -e destination.txt ]; then

cat source.txt > destination.txt

echo "Содержимое файла source.txt успешно скопировано в destination.txt"

else

echo "Один из файлов не существует. Пожалуйста, убедитесь, что оба файла существуют."

fi

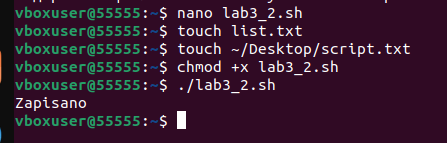
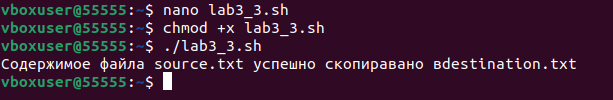


Рисунок 2. Скрипт для поиска файлов, начинающихся с "script" и записи их имен в list.txt

find ~/Desktop -type f -name "script\*" > list.txt

echo "Имена файлов, начинающихся с 'script', записаны в list.txt"

****Рисунок 3. Скрипт для выполнения скриптового файла, если он существует

if [ -e lab3\_1.sh ]; then

bash lab3\_1.sh

else

echo "Файл script.sh не существует."

fi

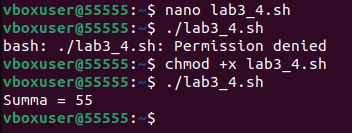
****

Рисунок 4. Скрипт для подсчета суммы от 1 до 10

sum=0

for ((i=1; i<=10; i++)); do

sum=$((sum + i))

done

echo "Сумма от 1 до 10 равна $sum"

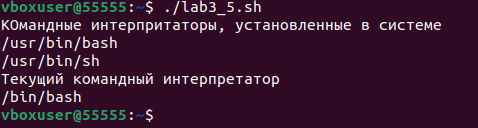
****

Рисунок 5. Поиск местонахождения командных интерпретаторов в Linux и определения текущего используемого интерпретатора

echo "Командные интерпретаторы, установленные в системе:"

which bash

which sh

which zsh

echo "Текущий командный интерпретатор:"

echo $SHELL

**Задание 2**

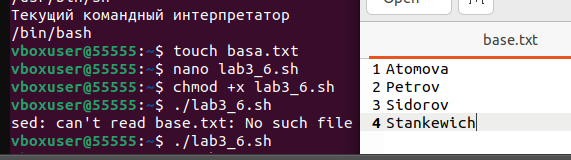


Рисунок 6. Замена фамилии в тексте первого примера:

sed -i 's/Nutrina/Stankewich/g' base.txt

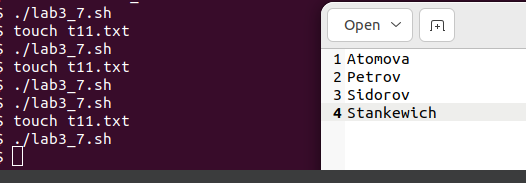


Рисунок 7. Cортировки файла, не меняя его собственного имени

sort base.txt > t11.txt

mv t11.txt base.txt



Рисунок 8. Подсчета числа одинаковых слов в двух файлах

sort base.txt > s1.txt

sort base2.txt > s2.txt

comm -12 s1.txt s2.txt | wc -l

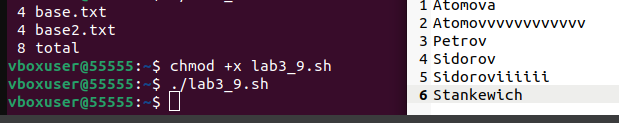


Рисунок 9. Cоздание отсортированного файла, содержащего слова из двух файлов, исключая общую часть одинаковых слов

cat base.txt base2.txt | sort | uniq > base3.txt

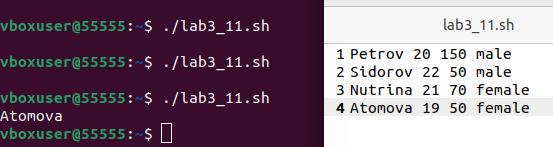


Рисунок 10. вывода фамилии самого молодого человека из файла base4.txt (второй столбец)

awk 'NR == 1 { youngest = $2 } NR > 1 && $2 < youngest { youngest = $2; name = $1 } END { print name }' base4.txt



Рисунок 11. Для вывода зарплаты самого молодого человека (третий столбец) из файла base4.txt

awk 'NR == 1 { youngest = $2; salary = $3 } NR > 1 && $2 < youngest { youngest = $2; salary = $3 } END { print salary }' base4.txt

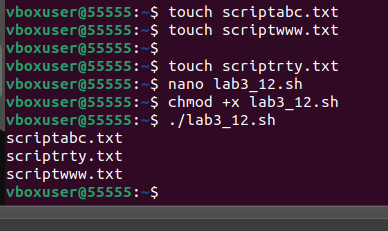


Рисунок 12. Вывод отсортированного списка имен файлов, начинающихся с "script"

ls | grep '^script' | sort

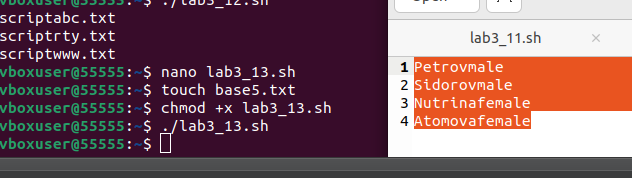


Рисунок 13. Удаление всех символов, кроме буквенных, в каждой строке файла

while IFS= read -r line; do

cleaned\_line=$(echo "$line" | tr -cd '[:alpha:]')

echo "$cleaned\_line"

done < inputfile.txt > outputfile.txt

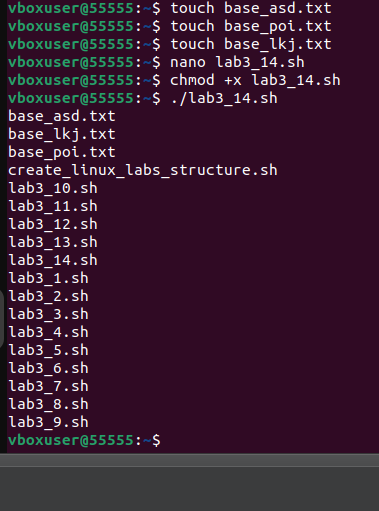


Рисунок 14. Вывода упорядоченного списка имен файлов, в которых содержится символ подчеркивания

ls | grep '\_' | sort

**Задание 3**

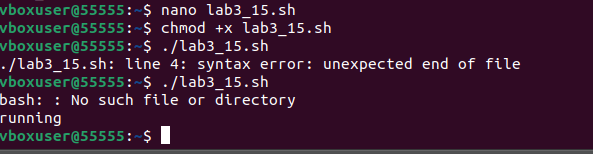


Рисунок 15. Скрипт, который запускает другой скрипт

script\_to\_run="lab\_1"

if [ -e "$script\_to\_run" ]; then

# Запуск скрипта

bash "$script\_to\_run"

echo "Скрипт успешно выполнен."

else

echo "Скрипт не найден."

Fi

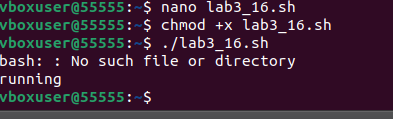


Рисунок 16. Расположение скрипта в одной из стантадртных директорий системы, и запуск его из домашней директории пользовател

script\_to\_run="~/Desktop/lab\_3 "

if [ -e "$script\_to\_run" ]; then

# Запуск скрипта

bash "$script\_to\_run"

echo "Скрипт успешно выполнен."

else

echo "Скрипт не найден."

Fi

**Контрольные вопросы**

**Что такое переменная окружения PATH?**

Переменная окружения PATH (Path) является одной из наиболее важных переменных окружения. Она определяет список директорий, в которых операционная система будет искать исполняемые файлы, когда вы выполняете команды из командной строки или скрипты.

**Что такое потоки данных, и какие стандартные потоки данных**

**существуют в системе? В какой системной директории располагаются их файлы?**

Потоки данных (или стандартные потоки) представляют собой основной механизм ввода и вывода

. Они представляют собой абстракцию для передачи данных между программами, устройствами и файлами. В системе существуют три стандартных потока данных:

stdin (стандартный входной поток): Этот поток предназначен для ввода данных в программу. Обычно он связан с клавиатурой, и программа может читать данные, которые вы вводите с клавиатуры, из этого потока. В системе файл для stdin не существует, так как данные поступают непосредственно от пользователя.

stdout (стандартный выходной поток): Этот поток предназначен для вывода данных из программы. Обычно он связан с экраном (терминалом), и программа выводит результаты своей работы в этот поток. В системе stdout ассоциирован с файлом /dev/stdout, который представляет собой специальное устройство, связанное с выводом на экран.

stderr (стандартный поток ошибок): Этот поток также предназначен для вывода данных, но он используется для сообщений об ошибках и другой отладочной информации, которая не является частью нормального вывода программы. Обычно stderr также связан с экраном. В системе stderr ассоциирован с файлом /dev/stderr, который представляет собой специальное устройство для вывода ошибок.

Стандартные потоки данных являются частью концепции "всё является файлом" в Unix-подобных системах, где даже устройства и потоки данных представлены как файлы в файловой системе.

Системные директории, где располагаются файлы stdin, stdout и stderr, зависят от конкретного дистрибутива Linux, но обычно они находятся в директории /dev. Таким образом:

/dev/stdin представляет стандартный входной поток (stdin).

/dev/stdout представляет стандартный выходной поток (stdout).

/dev/stderr представляет стандартный поток ошибок (stderr).