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 № 4

по дисциплине:” Системное программирование”

на тему:” Скрипты, диалоги, создание проекта”

Вариант 12

Выполнила**:** студент группы 10701222 Шкробот А.А.

Принял**:** ст. пр. Давыденко Н.В.

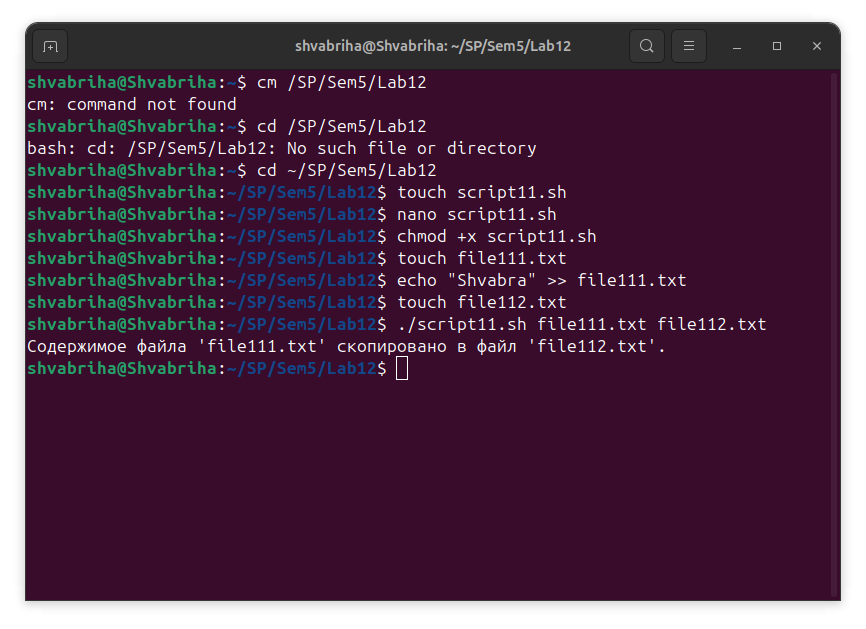
Минск 2024

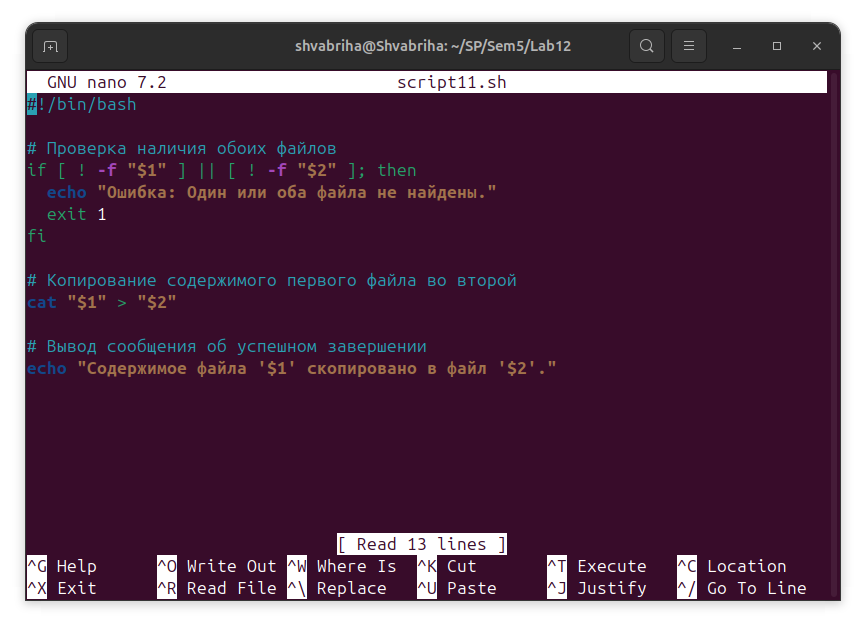
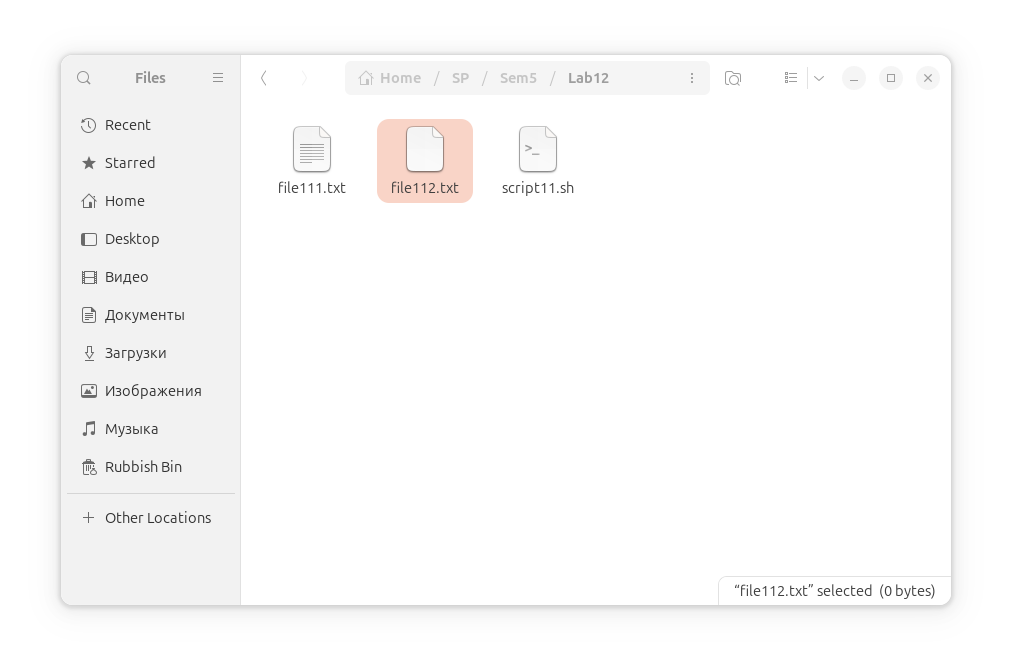
Цель работы

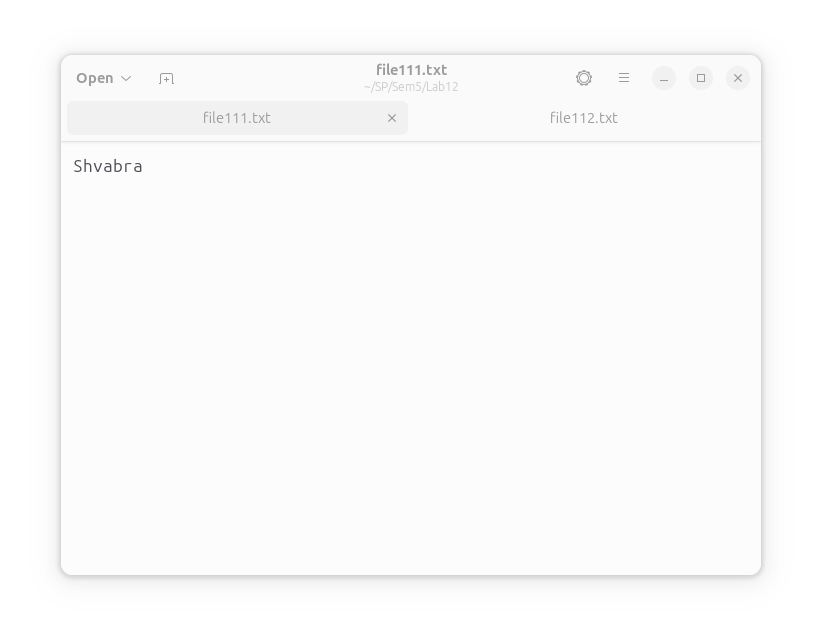
Закрепить на практике основы работы со сценариями командной оболочки и перенаправление ввода/вывода.

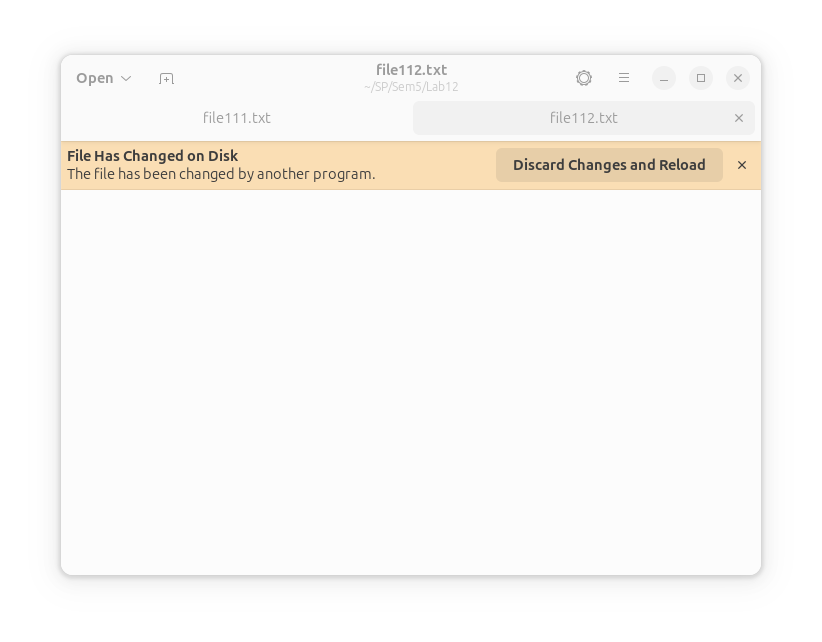
**Задание 1**

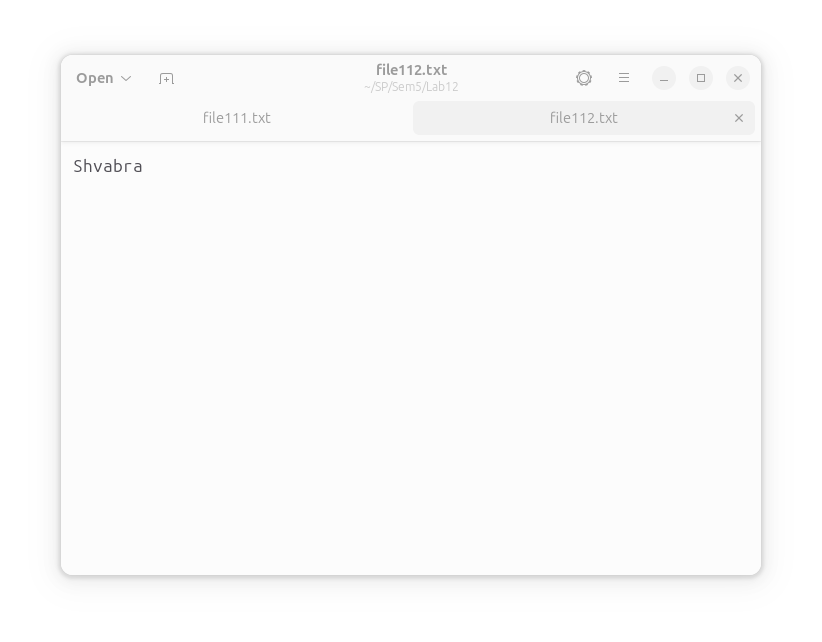
1. Написать скрипт, который записывает содержимое одного файла в другой и при этом оба файла существуют до операции. Дать два разных варианта выполнения.

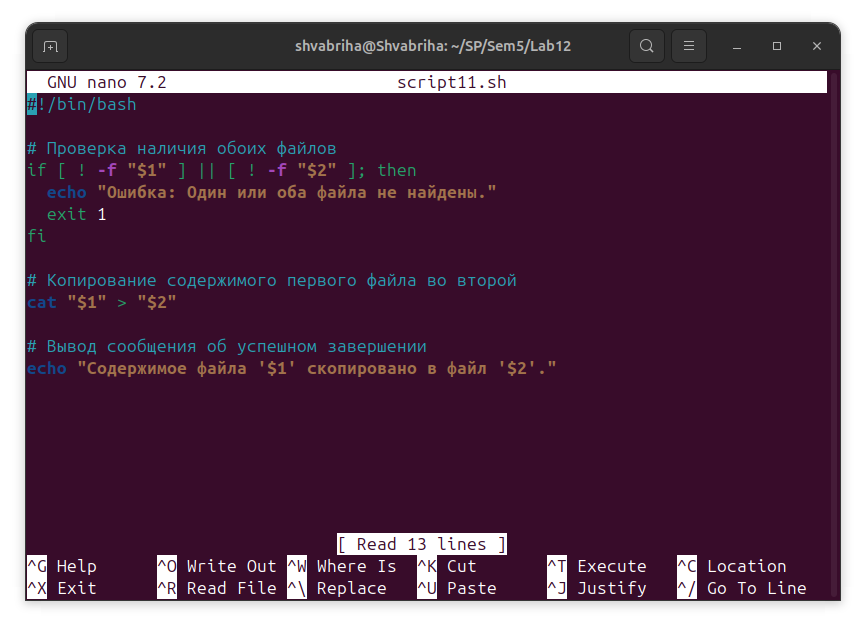




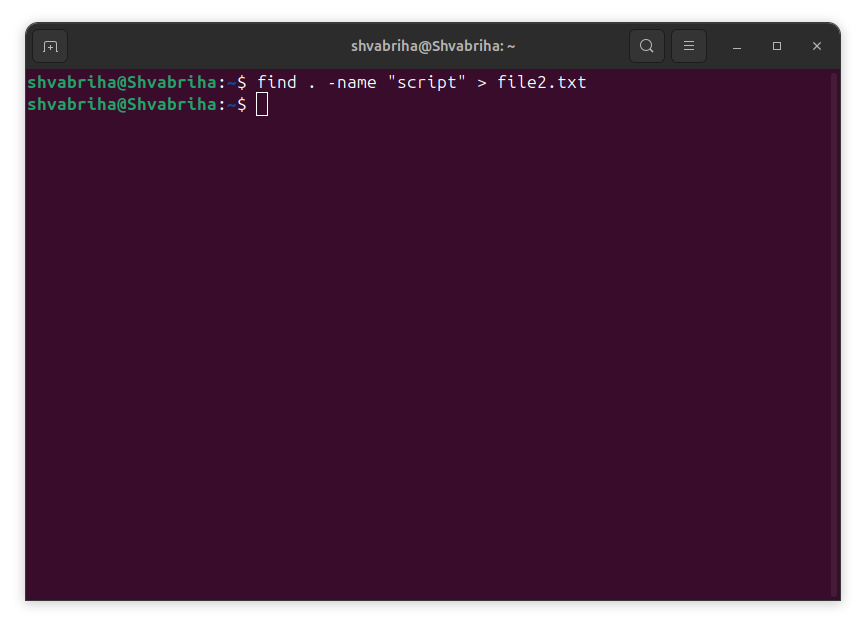


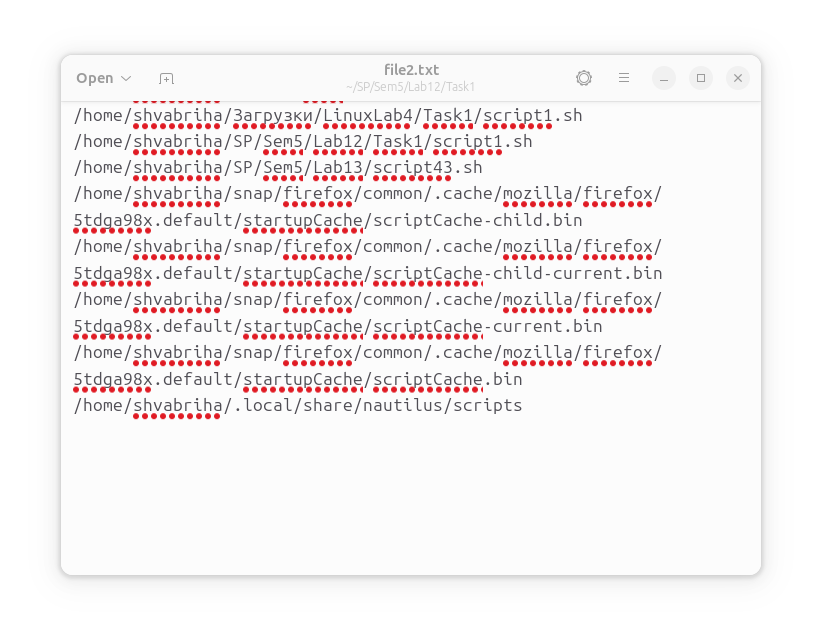




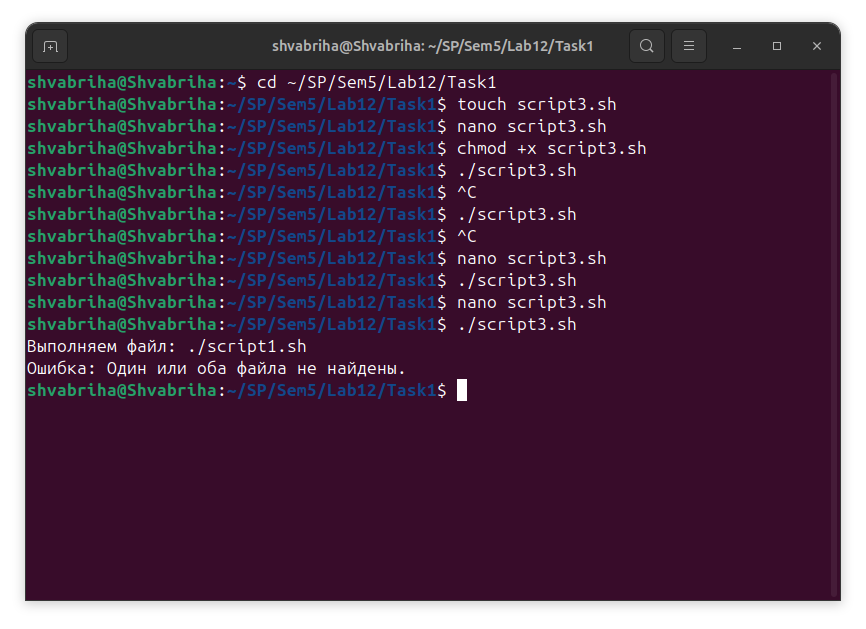


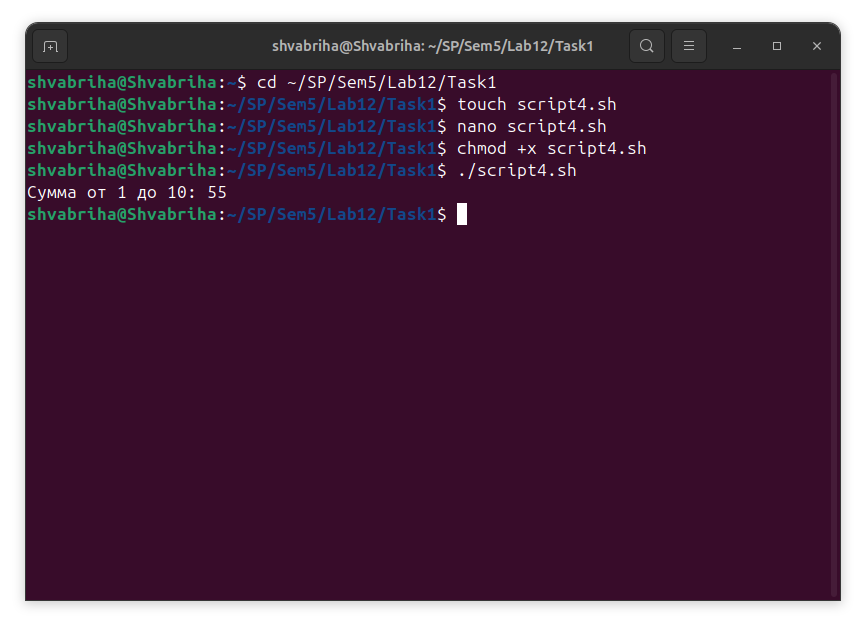
2. Найти все файлы, начинающиеся на слово script и записать их имена в файл list.txt.



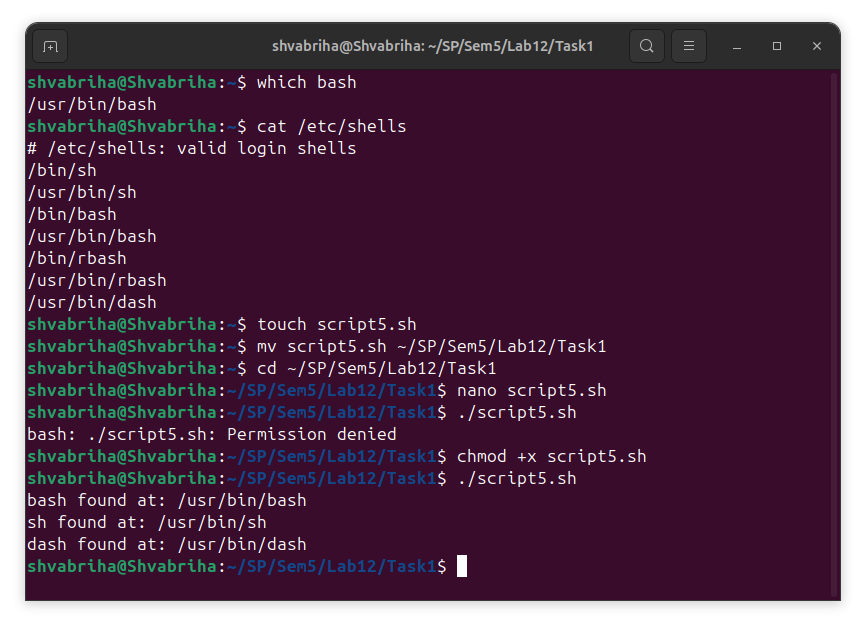


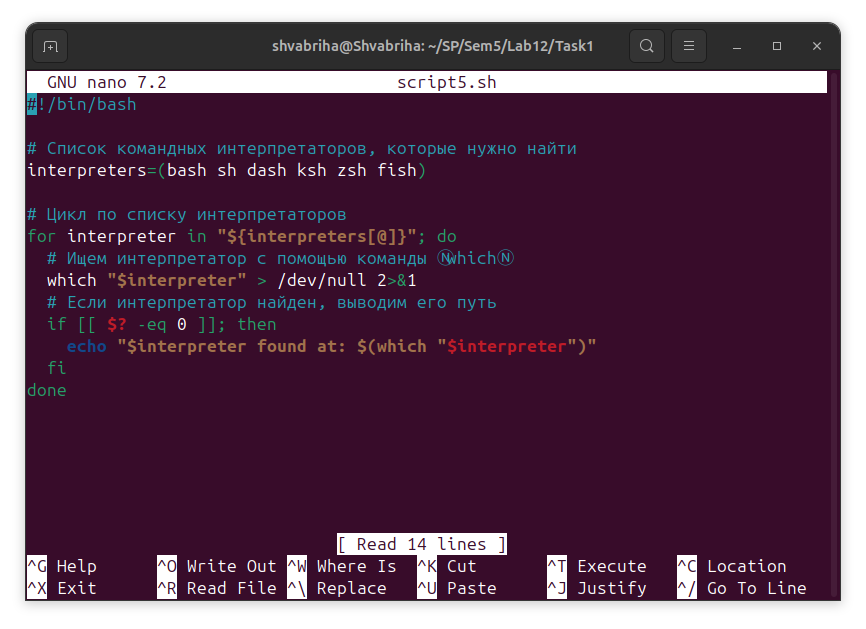
3. Написать скрипт, который проверяет, содержится ли скриптовый файл в директории, если да, то выполнить его.



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

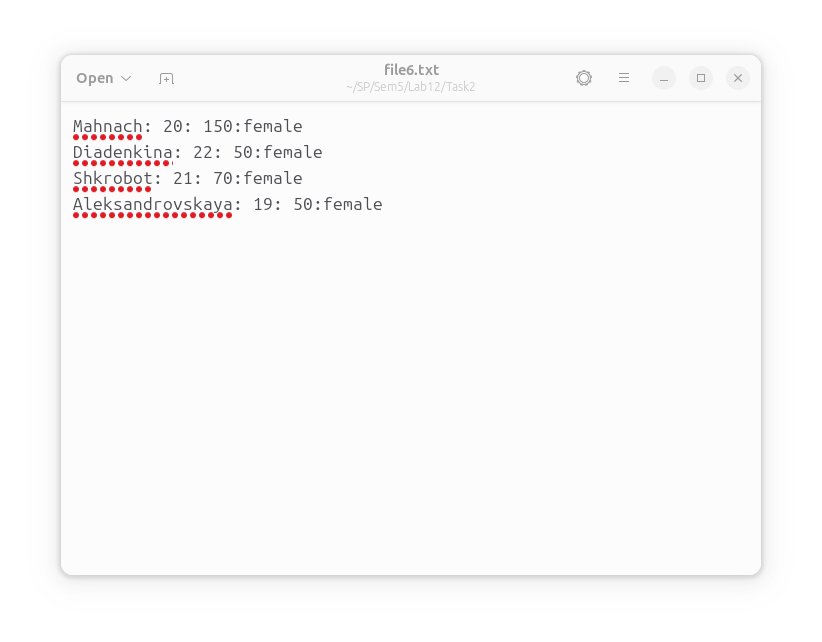
5. А какие командные интерпретаторы установлены в вашей системе? Напишите скрипт поиска местонахождения командных интерпретаторов вашей системы. Также определите какой интерпретатор используется сейчас в вашей системе?



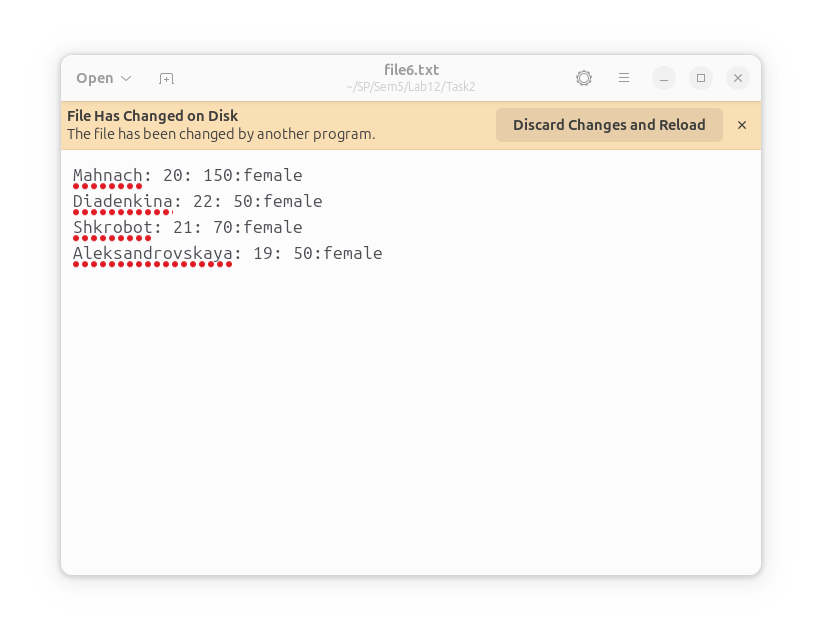
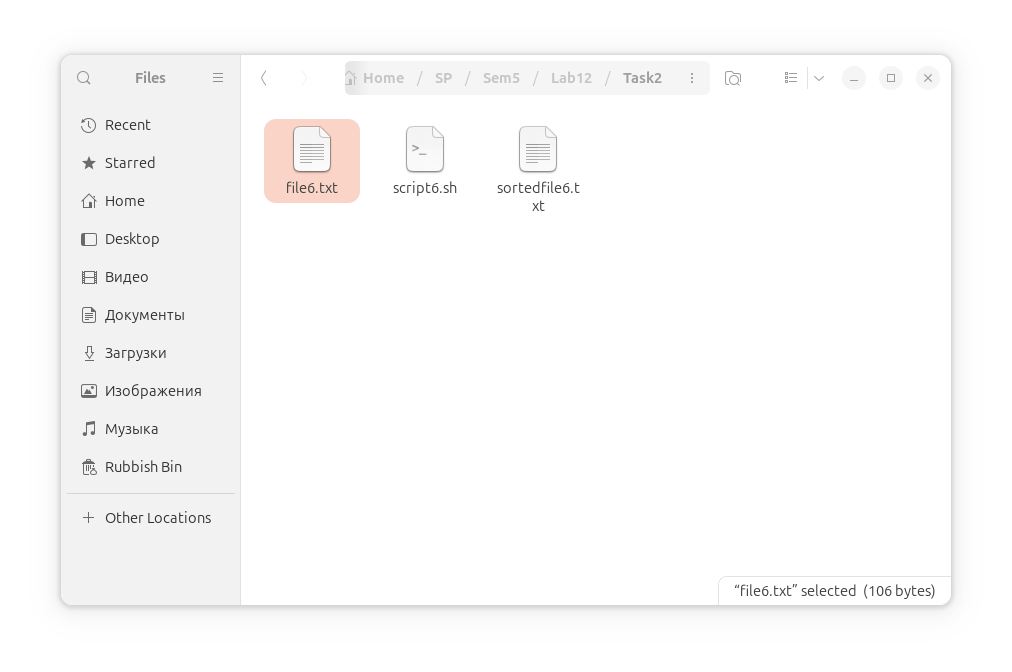


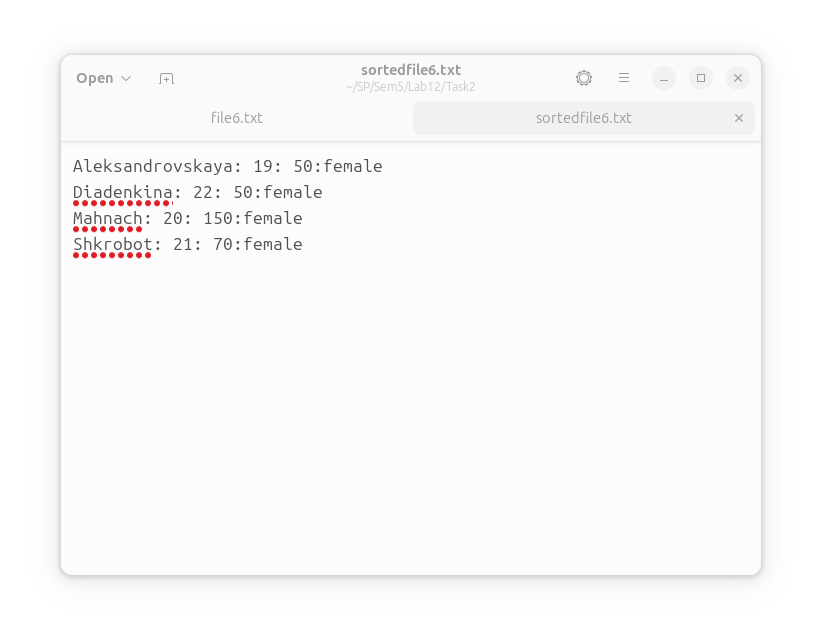
**Задание 2**

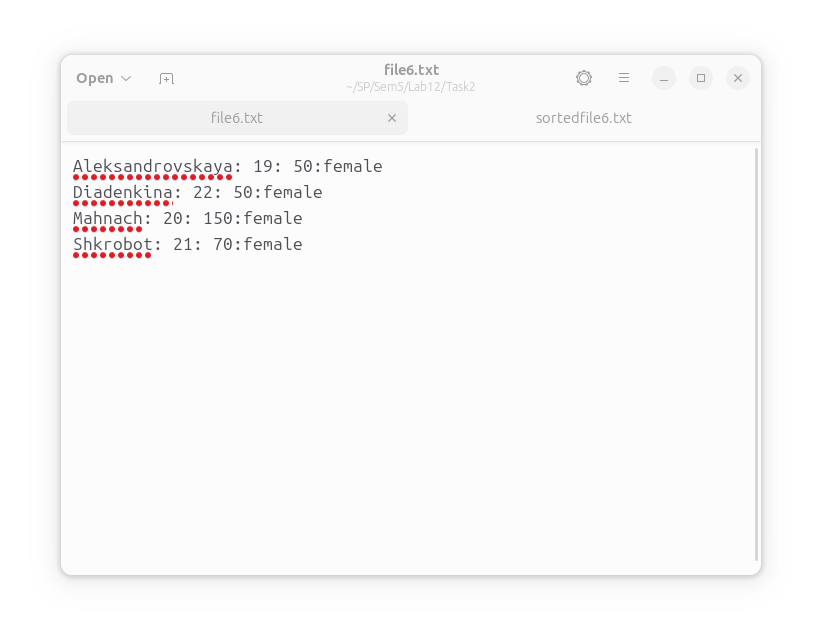
6. Используйте текст первого примера, заменить некоторые фамилии на фамилии членов вашей команды (например, Nutrina заменить на Stankewich).



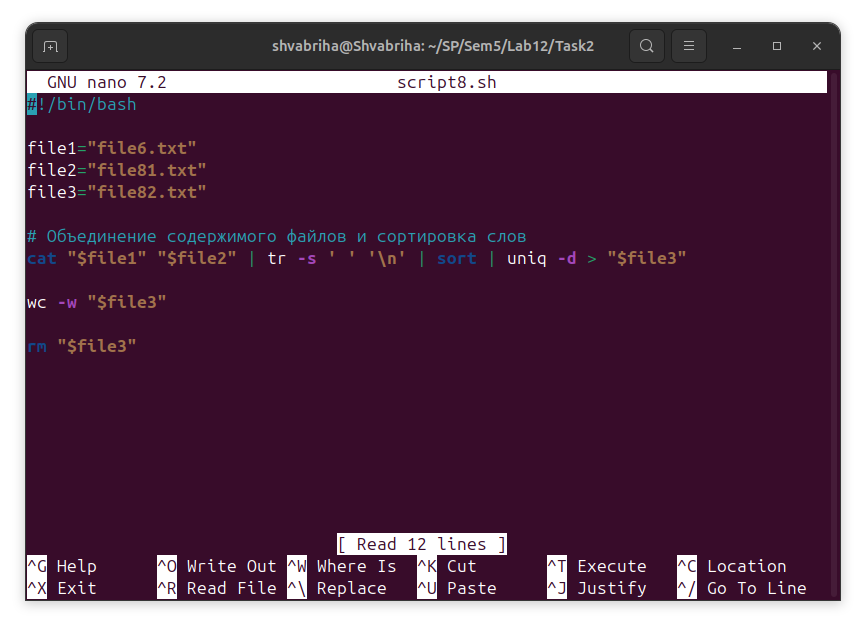
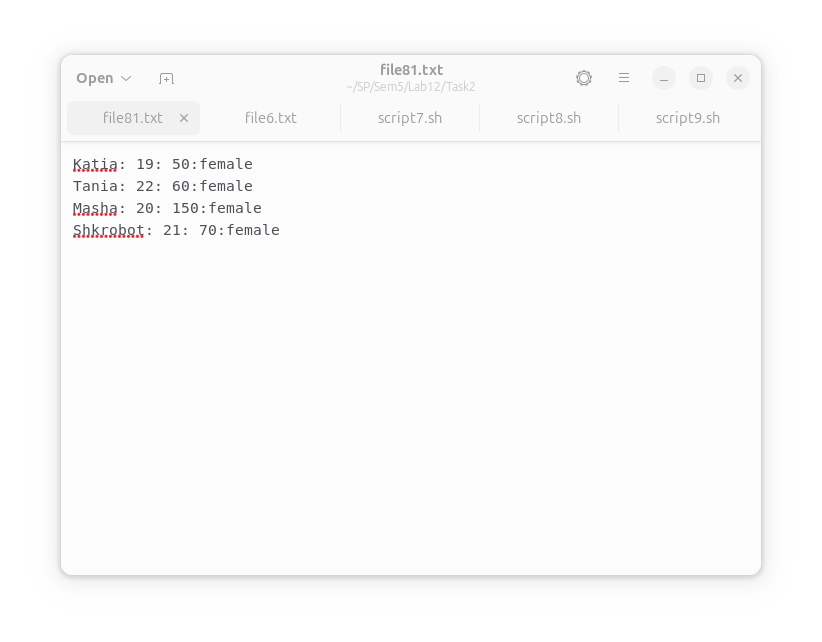
7. Отсортируйте данные файла так, чтобы он не поменял свое собственное имя. (Решение. Нужно предварительно создать промежуточный отсортированный файл t11.txt. Затем выполнить нужные команды).

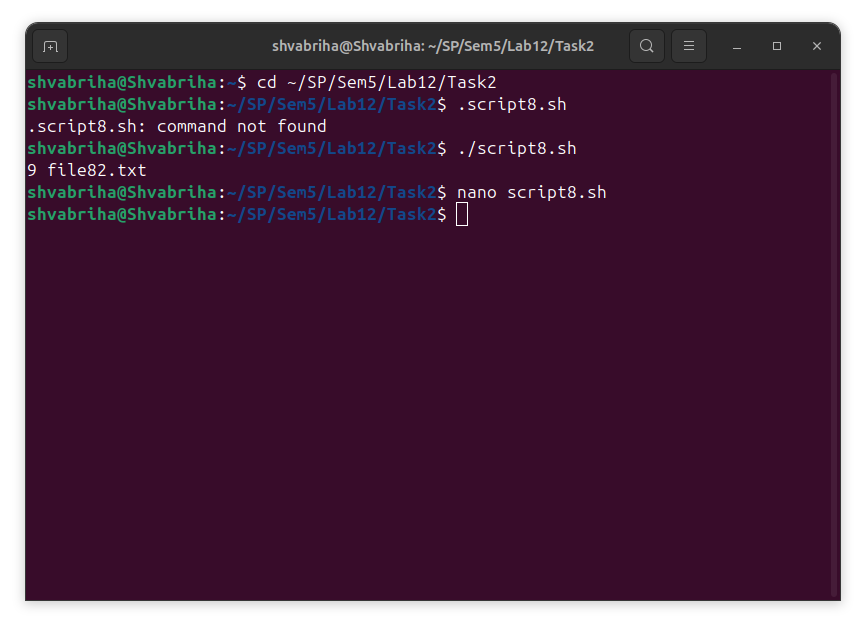




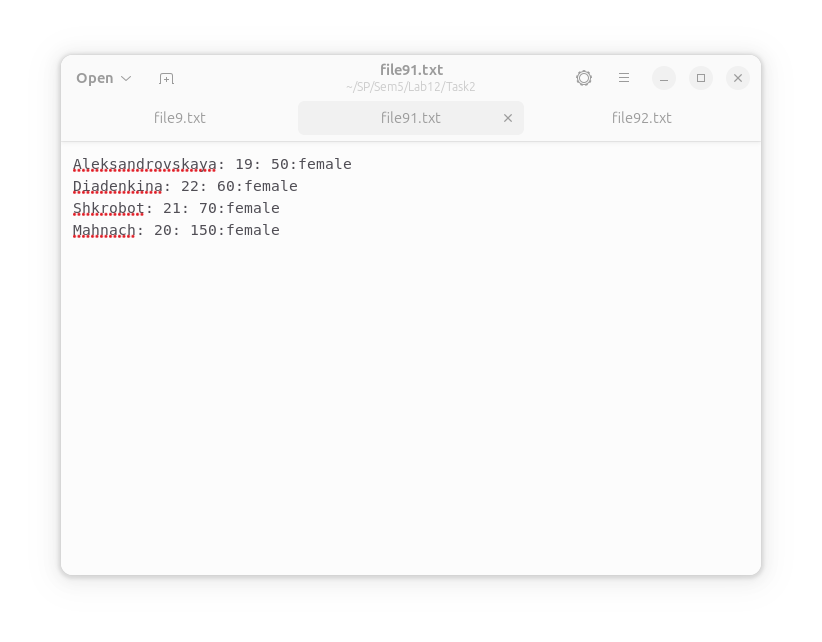


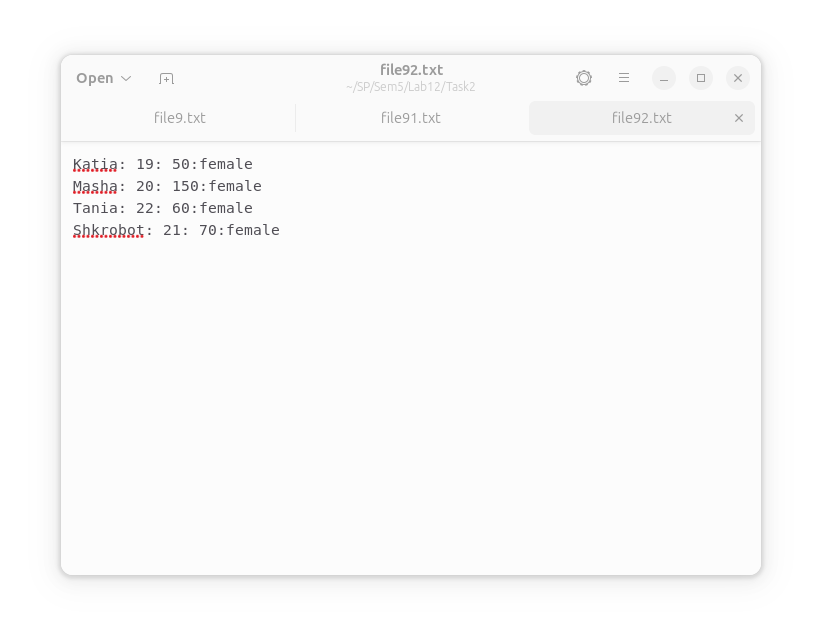
8. Подсчитайте число одинаковых слов в обоих файлах. Для подсчета числа слов в файле используйте команду: wc -w file1.

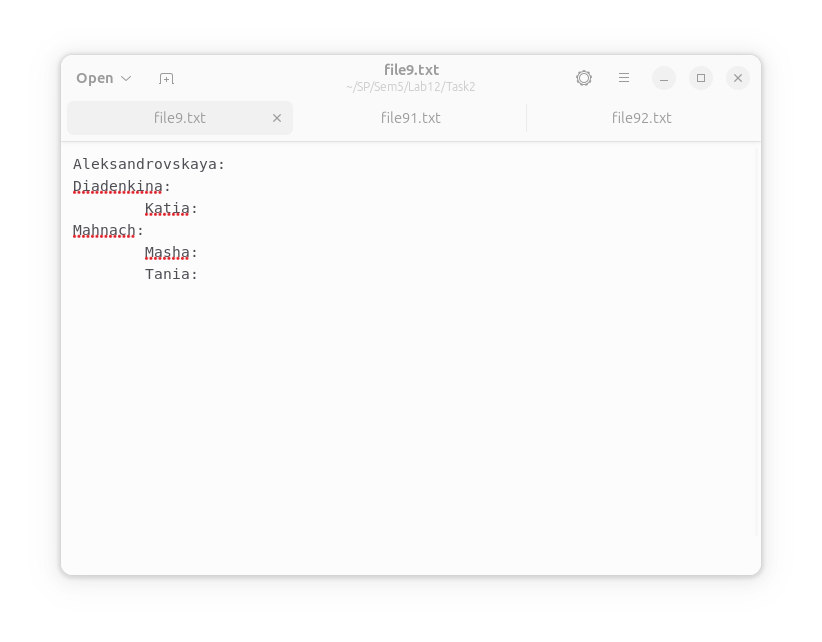




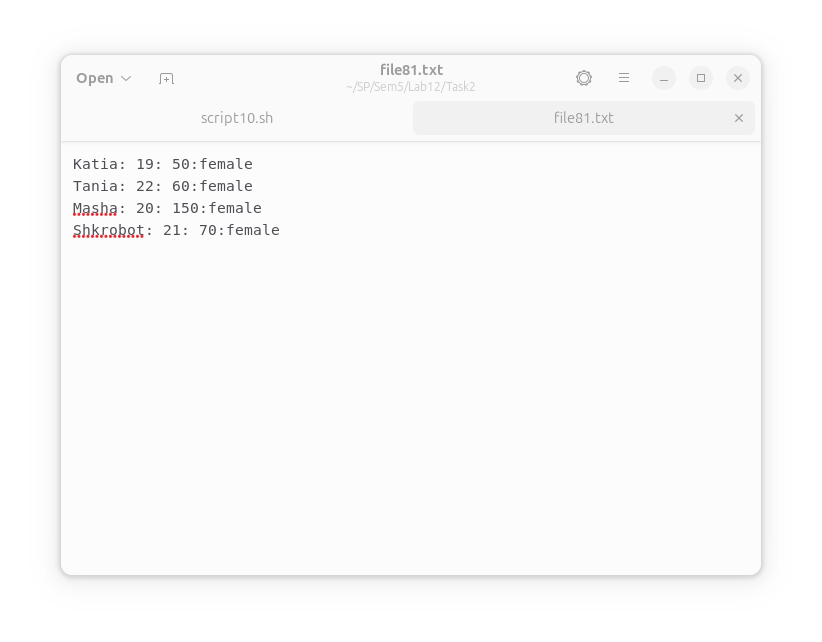
9. Написать скрипт, который создает отсортированный файл, содержащий слова из двух файлов, исключая их общую часть одинаковых слов.

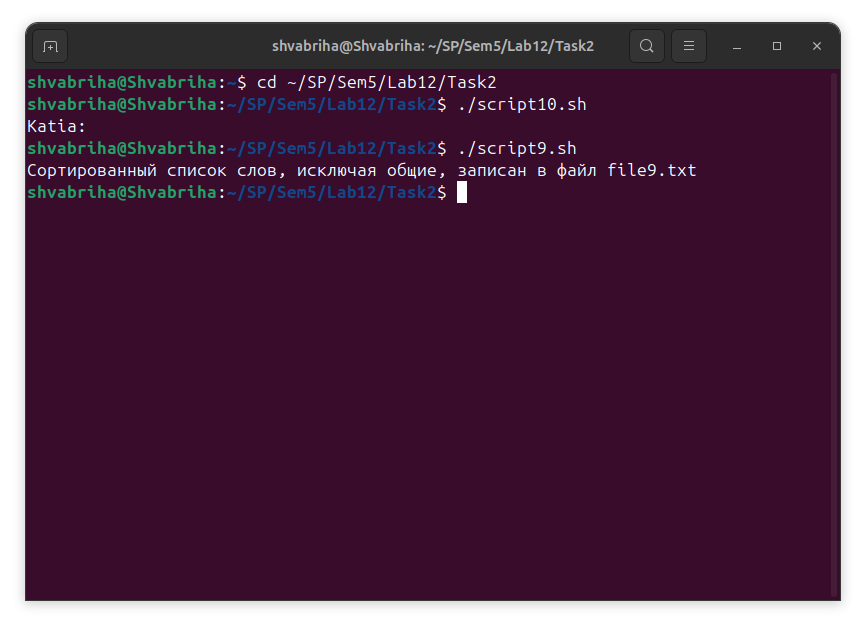


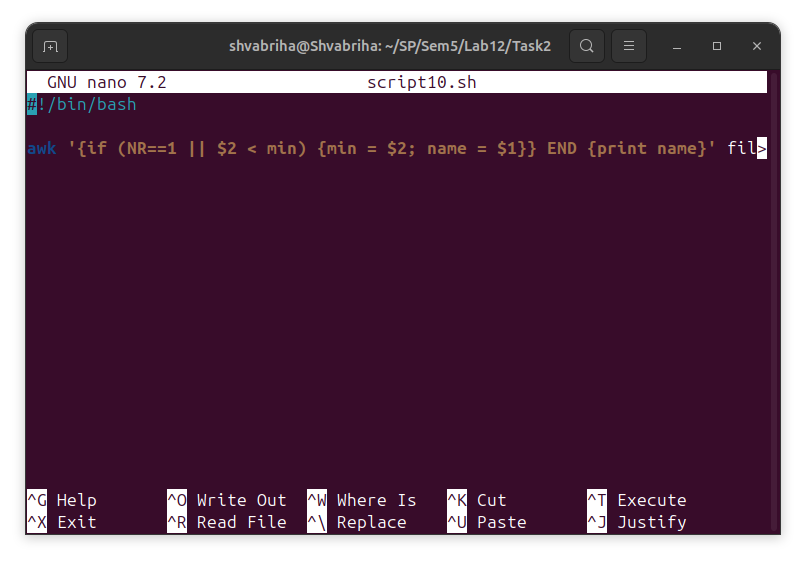




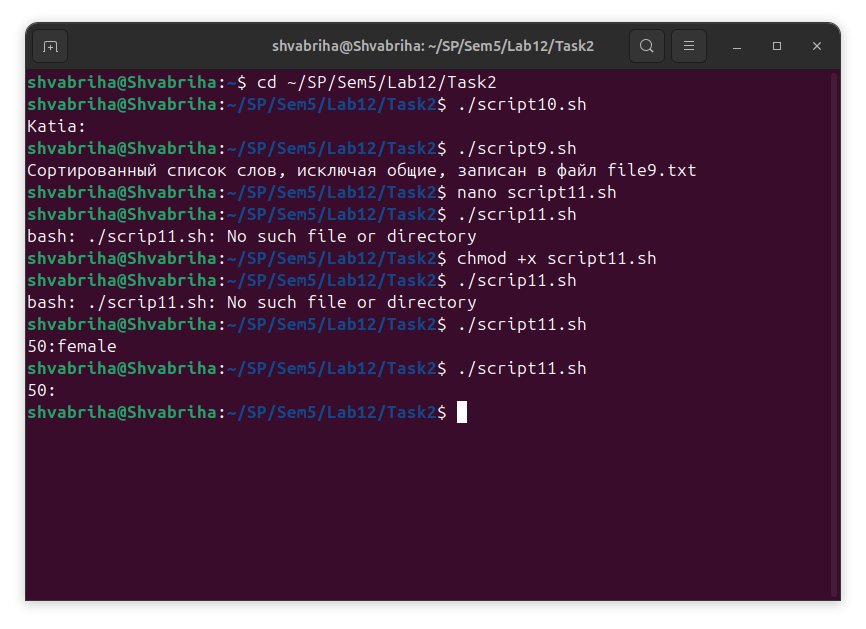
10.Вывести фамилию самого молодого человека (в файле basa.txt – это второй столбец).

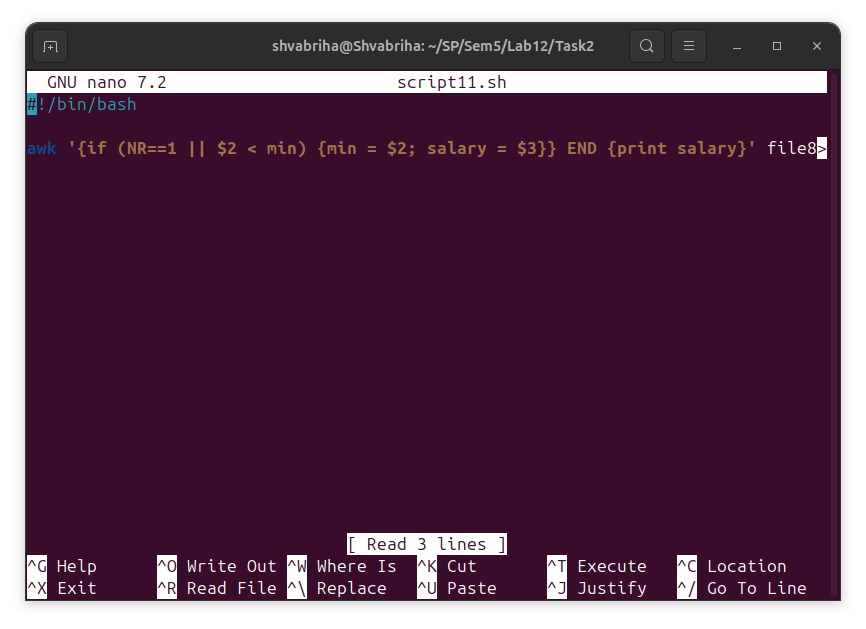




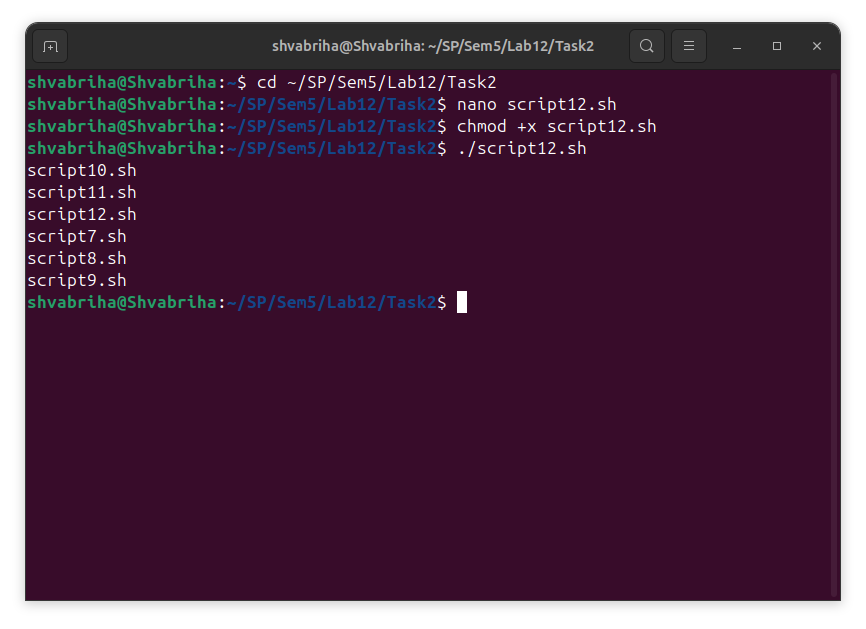


11.Вывести зарплату самого молодого человека (зарплата – третий столбец).

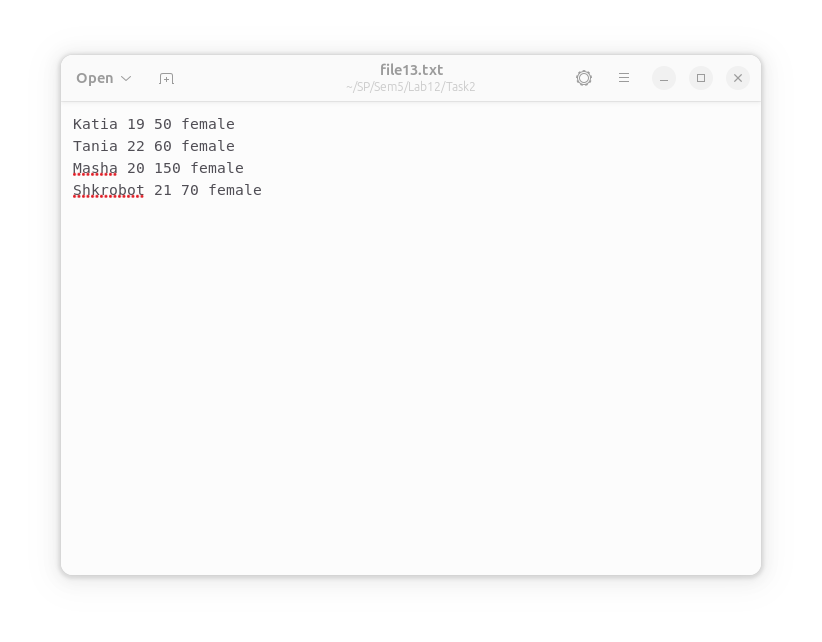
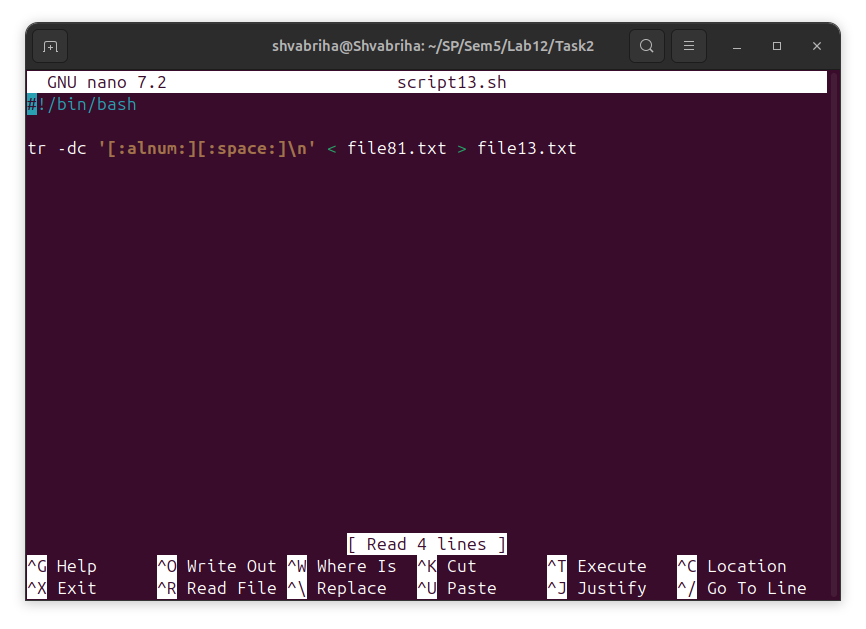
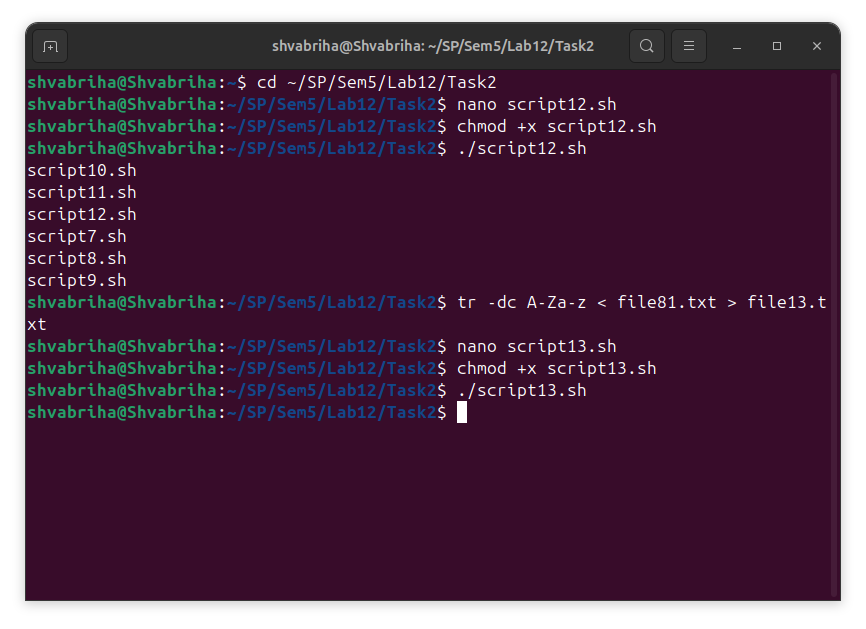




12.Вывести отсортированный список имен файлов, начальная часть имени есть script.



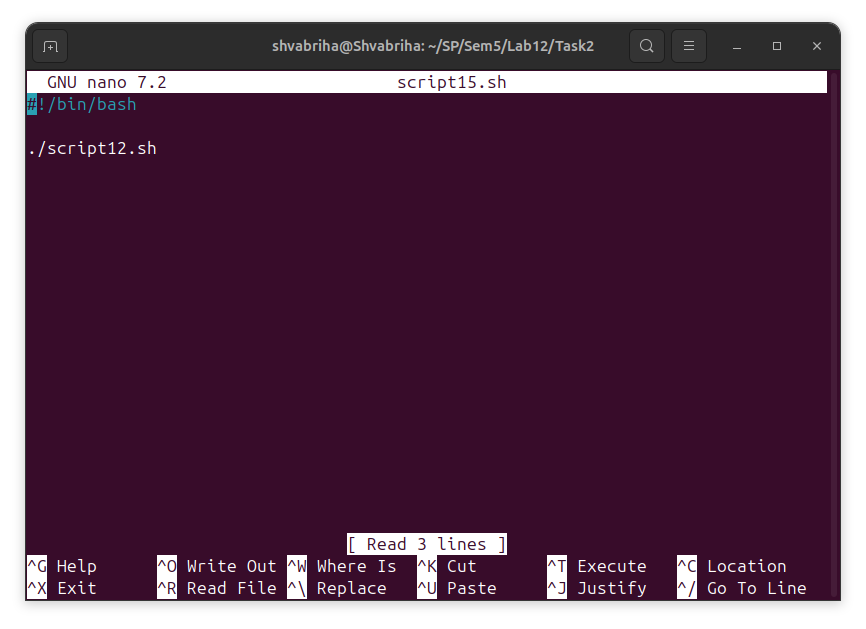
13.Написать скрипт, который в каждой строке файла оставляет только буквенные символы, а остальные символы выбрасывает.

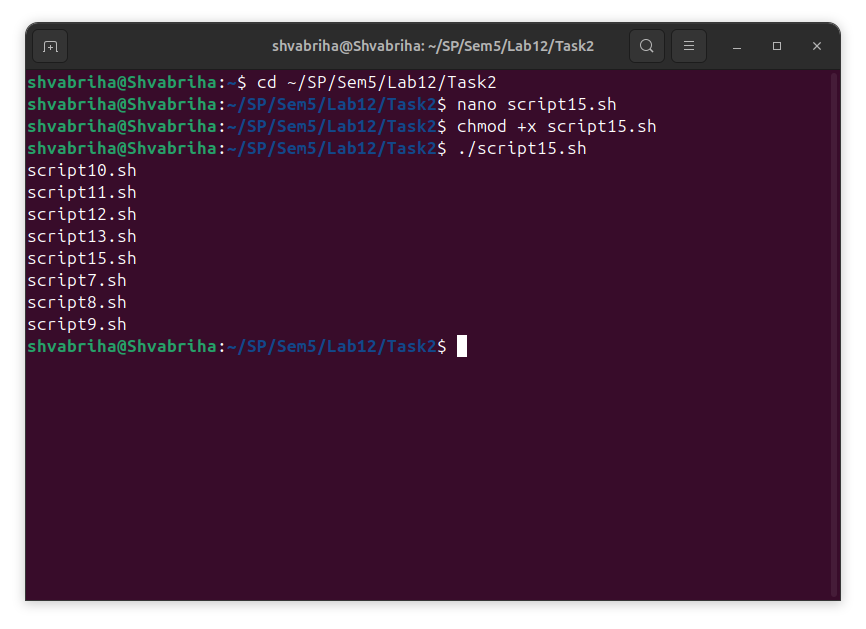


14.Вывести упорядоченный список имен файлов, в именах которых содержится символ подчеркивания, например, script\_sort.

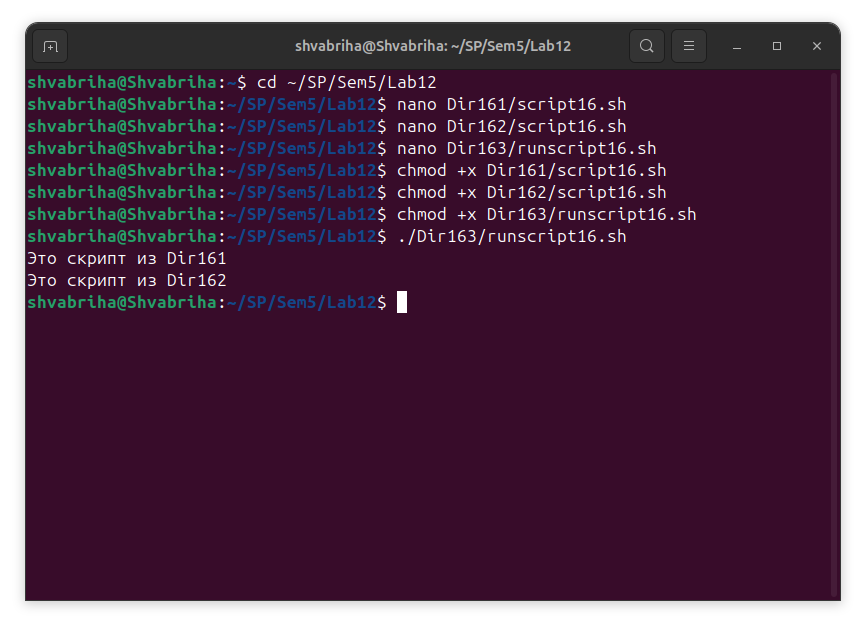
**Задание 3**

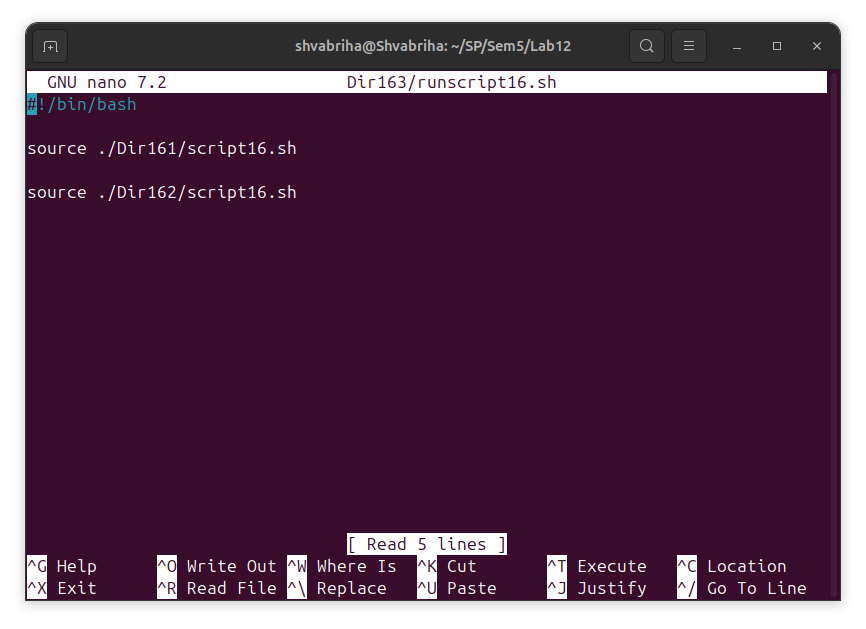
15.Напишите скрипт запуска другого скрипта из текущей директории.

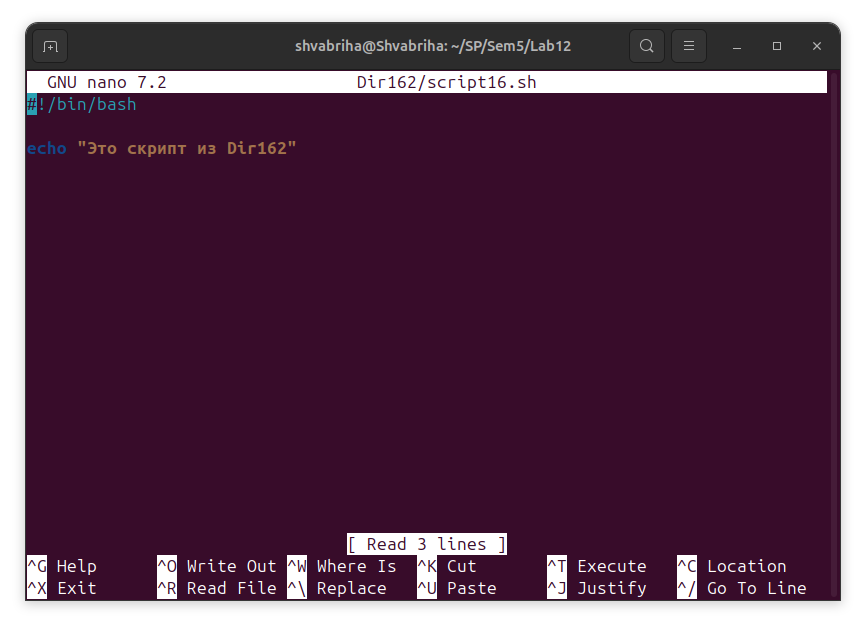


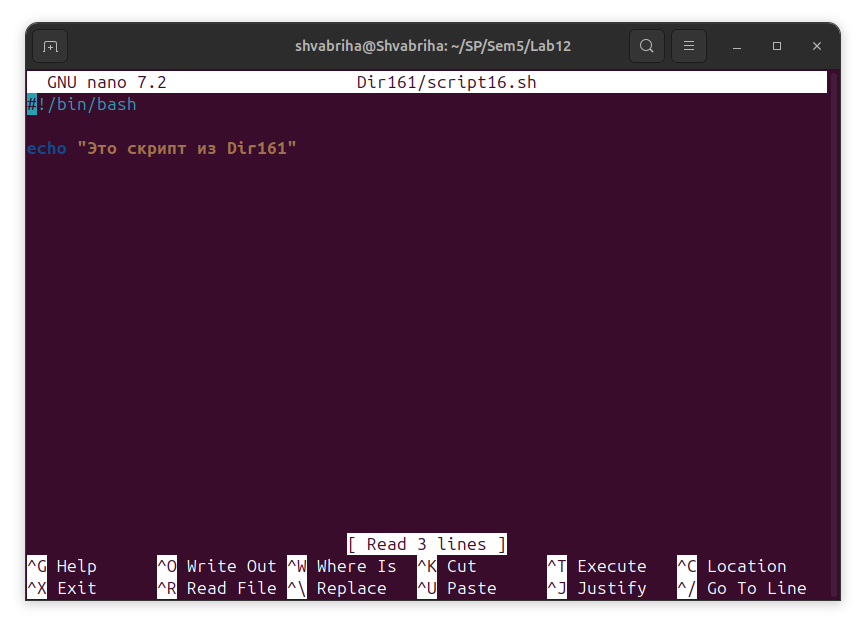


16.Создайте два скрипта с одинаковым именем, но разным текстом. Эти скрипты должны быть в разных директориях. В третьей директории создайте скрипт, запускающий два первых скрипта. При этом используйте команду sours. Объясните поведение программы и расскажите с какими проблемами вы столкнулись.

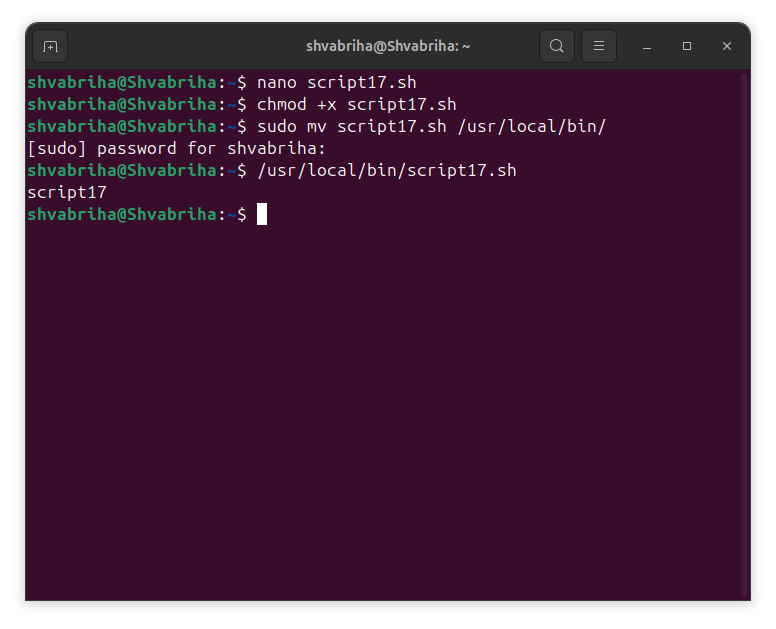


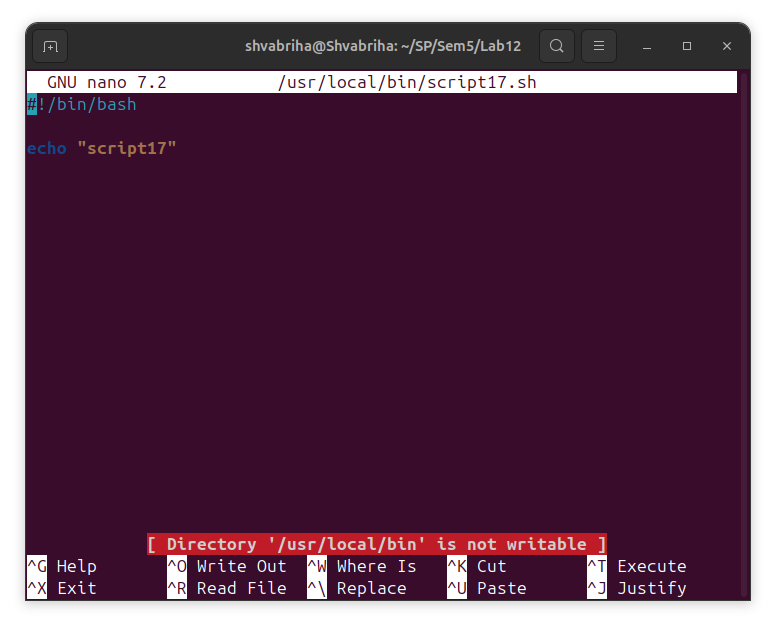




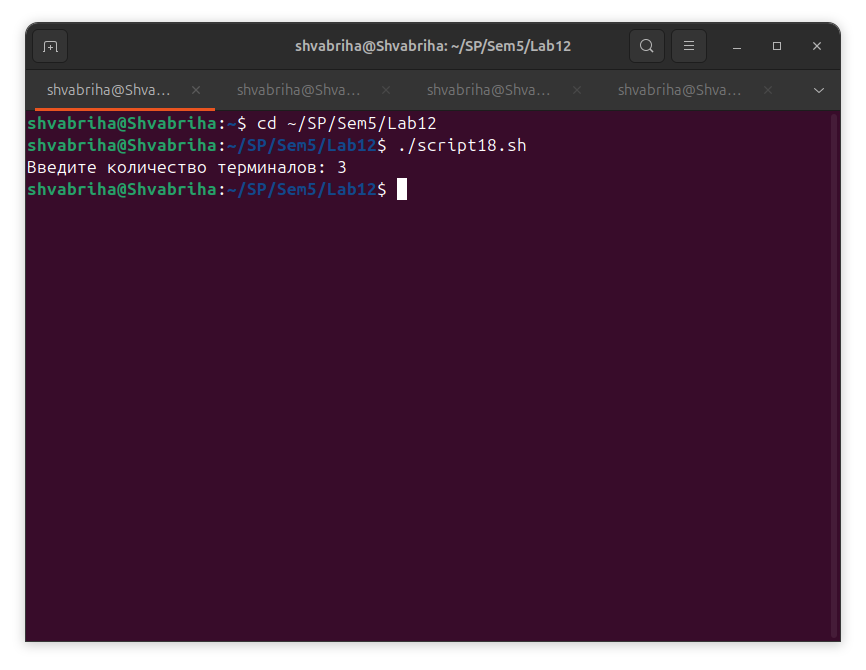


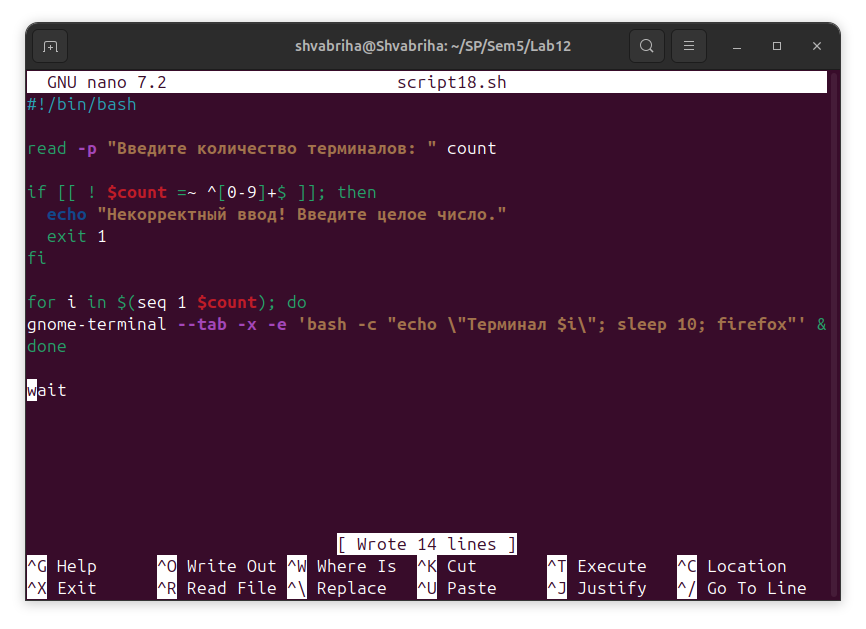
17.Расположите скрипт в одной из стантадртных директорий системы, и запустите его из домашней директории пользователя.

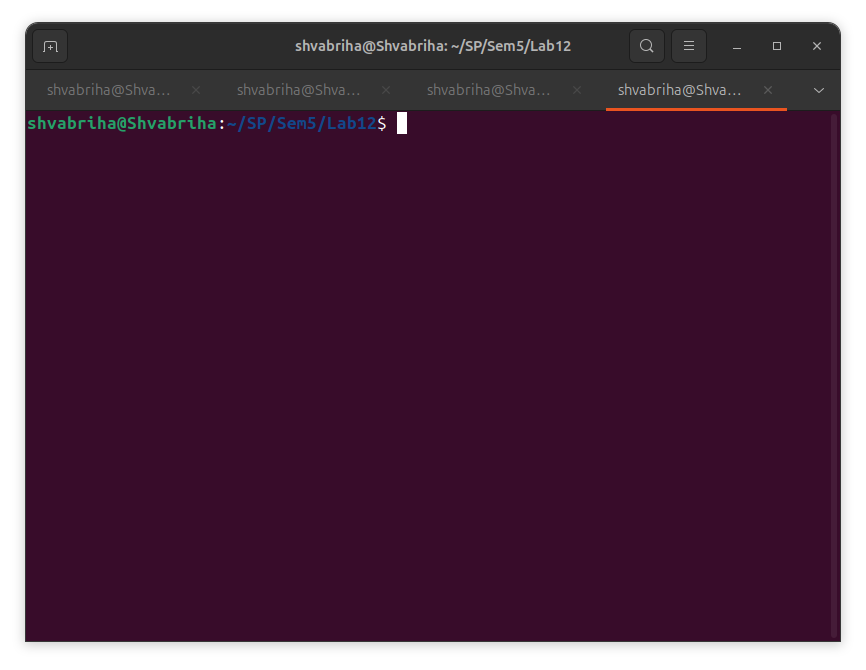




18.Напишите скрипт запускающий два и более терминалов отдельно (количество терминалов зависит от количества голов в команде). Количество терминалов должне вводиться интерактивно. Желательно чтобы в этих терминалах запускалисть какие нибудь программки, можно запустить браузер.







**Листинг**

Script1.sh

#!/bin/bash

# Проверка наличия обоих файлов

if [ ! -f "$1" ] || [ ! -f "$2" ]; then

echo "Ошибка: Один или оба файла не найдены."

exit 1

fi

# Копирование содержимого первого файла во второй

cat "$1" > "$2"

# Вывод сообщения об успешном завершении

echo "Содержимое файла '$1' скопировано в файл '$2'."

Script3.sh

#!/bin/bash

# Ищем файл с расширением .sh в текущем каталоге

find . -name "\*.sh" -print0 | while IFS= read -r -d $'\0' file; do

# Выполняем первый найденный файл

echo "Выполняем файл: $file"

"$file"

break # Прерываем цикл после выполнения первого файла

done

Script18.sh

#!/bin/bash

read -p "Введите количество терминалов: " count

if [[ ! $count =~ ^[0-9]+$ ]]; then

echo "Некорректный ввод! Введите целое число."

exit 1

fi

for i in $(seq 1 $count); do

gnome-terminal --tab 'bash -c "echo \"Терминал $i\"; sleep 10; firefox"' &

done

wait

Script4.sh

#!/bin/bash

sum=0

for i in $(seq 1 10); do

sum=$((sum + i))

done

echo "Сумма от 1 до 10: $sum"

Script5.sh

#!/bin/bash

# Список командных интерпретаторов, которые нужно найти

interpreters=(bash sh dash ksh zsh fish)

# Цикл по списку интерпретаторов

for interpreter in "${interpreters[@]}"; do

# Ищем интерпретатор с помощью команды ⓃwhichⓃ

which "$interpreter" > /dev/null 2>&1

# Если интерпретатор найден, выводим его путь

if [[ $? -eq 0 ]]; then

echo "$interpreter found at: $(which "$interpreter")"

fi

done

Script7.sh

#!/bin/bash

file="file6.txt"

sort "$file" > sorted"$file"

cat sorted"$file" > "$file"

rm sorted"$file"

Script8.sh

#!/bin/bash

file1="file6.txt"

file2="file81.txt"

file3="file82.txt"

# Объединение содержимого файлов и сортировка слов

cat "$file1" "$file2" | tr -s ' ' '\n' | sort | uniq -d > "$file3"

wc -w "$file3"

rm "$file3"

Script9.sh

#!/bin/bash

file1="file91.txt"

file2="file92.txt"

file3="file9.txt"

comm -3 <(cat "$file1" | tr -s ' ' '\n' | sort) <(cat "$file2" | tr -s ' ' '\n' | sort) > "$file3"

echo "Сортированный список слов, исключая общие, записан в файл file9.txt"

Script10.sh

#!/bin/bash

awk '{if (NR==1 || $2 < min) {min = $2; name = $1}} END {print name}' file81.txt

Script11.sh

#!/bin/bash

awk '{if (NR==1 || $2 < min) {min = $2; salary = $3}} END {print salary}' file81.txt

Script12.sh

#!/bin/bash

ls -1 script\* | sort

# task 14: ls -1 \*\_\* | sort

Script13.sh

#!/bin/bash

tr -dc '[:alnum:][:space:]\n' < file81.txt > file13.txt

Script15.sh

#!/bin/bash

./script12.sh

Script16.sh

#!/bin/bash

echo "Это скрипт из Dir161"

Script16.sh

#!/bin/bash

echo "Это скрипт из Dir162"

Script16.sh

#!/bin/bash

source ./Dir161/script16.sh

source ./Dir162/script16.sh

Контрольные вопросы

**Скрипты или сценарии командной оболочки**

6) Что такое переменная окружения PATH, и какую роль она играет в запуске скриптов?

Она позволяет исполнять скрипт по его имени и не указывая путь к нему. Path это список каталогов где Linux ищет исполняемые файлы.

Export PATH=$PATH:/home/user/scripts

**Перенаправление ввода/вывода и потоки данных**

10) Для чего используются команды-фильтры? Назовите некоторые из них.

Используются для обработки данных, поступающих из других команд или файлов для модификации формата содержимого или извлечения нужной информации

tr перевод(замена) символов

sort

uniq

wc подсчет слов

cat конкатенация