Национальный исследовательский университет «МЭИ»

Институт информационных и вычислительных технологий

Кафедра Вычислительных технологий

Отчет по лабораторной работе № 4

по курсу «Операционные системы»

Группа А-06-20

Выполнили:

Дереберя Т.М. (разработчик, менеджер)

Зинченко С.К. (аналитик, тестировщик)

Проверил:

Емельянов Д.М.

Github:

<https://github.com/FunnyBananchik/OS_Labs/tree/main/lab4>

Москва

2023

**Оглавление**

[**Постановка задачи** 3](#_Toc151023018)

[**Задание:** 3](#_Toc151023019)

[**Уточнения задания:** 3](#_Toc151023020)

[**Задание, переданное разработчику аналитиком:** 4](#_Toc151023021)

[**Инструкция по запуску программы:** 6](#_Toc151023022)

[**Код программы:** 7](#_Toc151023023)

[**task1.sh:** 7](#_Toc151023024)

[**task2.sh:** 8](#_Toc151023025)

[**task3.sh:** 9](#_Toc151023026)

[**task4.sh:** 10](#_Toc151023027)

[**task\_hub.sh:** 11](#_Toc151023028)

[**Тестирование программы:** 16](#_Toc151023029)

[**task\_hub.sh** 17](#_Toc151023030)

[**task1.sh** 20](#_Toc151023031)

[**task1\_test.sh** 20](#_Toc151023032)

[**task2.sh** 22](#_Toc151023033)

[**task2\_test.sh** 22](#_Toc151023034)

[**task3.sh** 24](#_Toc151023035)

[**task3\_test.sh** 24](#_Toc151023036)

[**task4.sh** 26](#_Toc151023037)

[**task4\_test.sh** 26](#_Toc151023038)

# **Постановка задачи**

## **Задание:**

Лабораторная работа №4 выполняется всей бригадой и требует разработки одного или нескольких bash/shell скриптов.  
Для этого участникам бригады необходимо распределить между собой роли менеджера, аналитика, разработчика и тестировщика.  
  
Преподаватель (далее - заказчик) выдвигает следующие функциональные требования к разрабатываемому \_Решению\_ под ОС Linux.  
1. В качестве предметной области и исходных данных выступает Файловая Система Преподавателя из Лабораторной работы №3.  
2. Базовое решение для успешной сдачи лабораторной работы должно предоставлять следующие возможности по заданному пользователем номеру группы:  
\* вывод имени студента с максимальным количеством пятерок/четверок/троек по тестам, вывод таких результатов;  
\* вывод имени студента, не сдавшего хотя бы один тест (с указанием номера теста);  
  
3. Для получения Отличной оценки (и потенциально дополнительных баллов) также необходимо в рамках решения реализовать:  
\* вывод по фамилии студента номеров пропущенных занятий;  
\* вывод топ-10 студентов с самыми длинными фамилиями;  
  
Требования могут уточняться менеджером или аналитиком бригады лично с заказчиком.  
  
В качестве отчетности, кроме демонстрации заказчику работоспособных скриптов, необходимо предоставить краткую документацию решения (в электроннм виде), отчет о проведенном тестировании (в электронном виде), а также документировать основные моменты разработки.

## **Уточнения задания:**

1. Выдавать данные в заданиях пункта 2 и первого задания пункта 3 отдельно по заданному предмету.
2. В заданиях второго пункта выводить всех студентов, получивших оценки\все номера тестов.
3. Под фамилией и именем в условии задания имеется в виду полное латинское ФИО
4. Задания 3 пункта выполняются по всем группам.

## **Задание, переданное разработчику аналитиком:**

1. Создать скрипт для выполнения задачи: «Вывод имени студента с максимальным количеством пятёрок/четвёрок/троек по тестам»
   1. В качестве входных параметров в скрипт должны передаваться:
      1. Название предмета.
      2. Номер группы.
      3. Искомая оценка.
   2. Поиск будет выполняться по номеру группы и искомой оценке в файлах labfiles/”Название предмета”/tests/\*.
   3. После выполнения скрипта должны быть выведены максимальное количество полученных требуемых оценок и полные латинские ФИО всех студентов с максимальным количеством искомых оценок в алфавитном порядке.
2. Создать скрипт для выполнения задачи: «Вывод имени студента, не сдавшего хотя бы один тест (с указанием номера теста)».
   1. В качестве входных параметров в скрипт должны передаваться:
      1. Название предмета.
      2. Номер группы.
   2. Поиск будет выполняться по номеру группы в файлах labfiles/”Название предмета”/tests/\*.
   3. После выполнения скрипта должны быть выведены полные латинские ФИО всех студентов, не сдавших хотя бы один тест с номерами всех несданных тестов, в алфавитном порядке.
3. Создать скрипт для выполнения задачи: «Вывод пропущенных занятий по фамилии студента»
   1. В качестве входных параметров в скрипт должны передаваться:
      1. Название предмета.
      2. Фамилия студента
   2. Поиск будет выполняться по фамилии студента в файлах labfiles/”Название предмета”/\*-attendance
   3. После выполнения скрипта должны быть выведены номера пропущенных занятий, через запятую.
4. Создать скрипт для выполнения задачи: «Вывод топ-10 студентов с самыми длинными фамилиями»
   1. В качестве входных параметров в скрипт должны передаваться:
      1. Номер группы (не обязательный).
   2. Поиск будет выполняться в файлах students/groups/”Номер группы” (\*, если он не указан).
   3. После выполнения скрипта должен быть выведен нумерованный список топ-10 самых длинных фамилий в группе (среди всех групп, если конкретная не указана).
5. Скрипт-хаб, в котором в вечном цикле будет осуществляться выбор желаемой задачи (с возможностью выхода), также в этом скрипте должен быть осуществлён ввод требуемых входных значений и обработка ввода некорректных данных.

# **Инструкция по запуску программы:**

Поместить файлы task1.sh, task2.sh, task3.sh, task4.sh и task\_hub.sh в одну директорию с папкой labfiles.

Запустите хаб-скрипт командой ./task\_hub.sh

Следуйте инструкциям, выводимым в консоль.

# **Код программы:**

## **task1.sh:**

#!/bin/bash

grep -i -h "$2" labfiles/$1/tests/TEST\* | grep "${3}$" > grep.txt

if [ ! -s grep.txt ]

then

echo "Нет людей, получивших оценку $3"

else

declare -A students

for n in $(cat grep.txt)

do

p=$(echo $n | awk -F ";" '{print $2}')

students[$p]=$(grep -c "$p" grep.txt)

done

for value in "${students[@]}"

do

echo $value >> mas.txt

done

max=$(sort -r mas.txt | head -1)

echo "Максимальное кол-во оценок $3 = $max"

for student in "${!students[@]}"

do

if ((${students[$student]} == max))

then

echo $student >> res.txt

fi

done

cat res.txt | sort

rm res.txt

fi

rm -f grep.txt mas.txt

## **task2.sh:**

#!/bin/bash

for i in {1..4}

do

if [ ! -z $(find "labfiles/$1/" -name "TEST-$i") ]

then

grep -i "$2" labfiles/$1/tests/TEST-$i | grep 2$ | awk -F ";" '{print $2}' | uniq > grep$i.txt

fi

done

cat grep\*.txt | sort | uniq > grep.txt

if [ -z $(head -1 grep.txt) ]

then

echo "Все студенты сдали тесты"

else

for student in $(cat grep.txt)

do

str="${student}"

for i in {1..4}

do

count=$(grep -c ${student} grep$i.txt)

if ((count != 0))

then

str="${str} $i"

fi

done

echo "${str}"

done

fi

rm -rf grep\*.txt

## **task3.sh:**

#!/bin/bash

data=$(grep -i -h "$2" labfiles/$1/\*-attendance | awk -F " " '{print $2}')

famil=$(grep -i -h "$2" labfiles/$1/\*-attendance | awk -F " " '{print $1}')

echo "$famil"

i=0

kol=0

while [ $i -lt ${#data} ]

do

if [ "${data:i:1}" == "0" ]

then

((kol++))

if [ $kol -eq 1 ]

then

echo 'Номера пропущенных занятий'

echo -n "$((i+1))"

elif [ $kol -gt 1 ]

then

echo -n ", $((i+1))"

fi

elif [ $kol -eq 0 ] && [ $i -eq $((${#data}-1)) ]

then

echo -n 'Нет пропущенных занятий'

fi

((i++))

done

echo ''

## **task4.sh:**

#!/bin/bash

if [ -z $1 ]

then

grep -h . labfiles/students/groups/\* > grep.txt

else

grep -h . labfiles/students/groups/$1 > grep.txt

fi

cat grep.txt | awk '{ print 30-length, $0 }' | sort -k 1 -n -k 2 |uniq |cut -d" " -f2- | head -10 > res.txt

n=$(grep -h -c . res.txt)

i=0

while [ $i -lt $n ]

do

echo -e "$((i+1)).\t$(tail -$((n-i)) res.txt | head -1)"

((i++))

done

rm -f grep.txt res.txt

## **task\_hub.sh:**

#!/bin/bash

#Functions

function Prov\_SubjectName(){

resalt1=0

while [ $resalt1 -ne 1 ]

do

echo 'Введите название предмета'

read SubjectName

p=$(find . -type d -name "$SubjectName")

if [ ! -z $p ]

then

resalt1=1

else

echo 'Неверное название предмета'

echo ' '

resalt1=0

fi

done

}

function Prov\_Ngroup(){

unset result1

while [ -z $result1 ]

do

echo 'Введите номер группы в формате "A-\*\*-\*\*"'

read Ngroup

for file in labfiles/$SubjectName/tests/TEST-\*

do

prov=$(grep -i -h -c "$Ngroup" $file)

if [ $prov -ne 0 ]

then

result1=1

break

fi

done

if [ $prov -eq $(grep -h -c "" $file) ]

then

unset result1

fi

if [ -z $result1 ]

then

echo 'Неверный номер группы'

echo ' '

fi

done

}

#main

unset flag

while [ -z $flag ]

do

echo 'Выберите задание: '

echo '1 - Вывод имени студента с максимальным количеством указанных оценок'

echo '2 - Вывод имени студента, не сдавшего хотя бы один тест (с номером теста)'

echo '3 - Вывод номеров пропущенных занятий по фамилии студента'

echo '4 - Вывод ТОП-10 студентов с самыми длинными ФИО'

read vibor

if [ "$vibor" == "1" ] # task1

then

Prov\_SubjectName

Prov\_Ngroup $SubjectName

unset flag1

while [ -z $flag1 ]

do

echo 'Введите искомую оценку'

read Score

if [ $Score != 3 ] && [ $Score != 4 ] && [ $Score != 5 ]

then

echo 'Неверная оценка'

echo ' '

else

flag1=1

fi

done

./task1.sh $SubjectName $Ngroup $Score

elif [ "$vibor" == "2" ] # task2

then

Prov\_SubjectName

Prov\_Ngroup $SubjectName

./task2.sh $SubjectName $Ngroup

elif [ "$vibor" == "3" ] # task3

then

Prov\_SubjectName

unset flag1

while [ -z $flag1 ]

do

echo 'Введите ФИО студента'

read FIO

for filename in labfiles/$SubjectName/\*-attendance

do

prov=$(grep -i -h -c "$FIO" $filename)

if [ $prov -ne 0 ]

then

flag1=1

break

fi

done

if [ -z $flag1 ]

then

echo 'Неверное ФИО студента '

echo ' '

fi

done

./task3.sh $SubjectName $FIO

elif [ "$vibor" == "4" ] # task4

then

unset flag3

while [ -z $flag3 ]

do

echo 'Введите номер группы или ENTER (если хотите искать по всем группам)'

read Ngroup

if [ ! -z $Ngroup ]

then

if [ ${Ngroup:0:1} == "a" ]

then

Ngroup="A${Ngroup:1}"

fi

for file in labfiles/students/groups/\*

do

if [ $file == "labfiles/students/groups/$Ngroup" ]

then

flag3=1

break

fi

done

if [ -z $flag3 ]

then

echo 'Неверный номер группы'

echo ' '

fi

else

flag3=1

fi

done

./task4.sh $Ngroup

else

echo 'Неправильный номер задания'

fi

echo ' '

echo 'Хотите ещё раз выбрать задание?'

echo 'Enter - да'

echo 'Любой символ - нет'

echo ' '

read flag

done

# **Тестирование программы:**

Для тестирования программы на краевых ситуациях была добавлена директория предмета криптоварение и тестовая группа A-F-K:

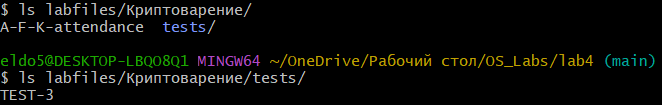
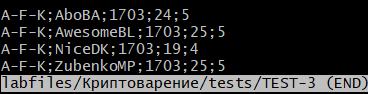


Рисунок 1 – просмотр созданной директории криптоварение

  
Рисунок 2 – добавление группы A-F-K в файл TEST-3 предмета криптоварение

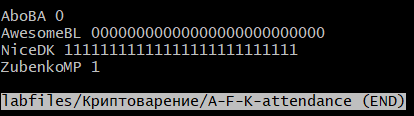


Рисунок 3 – добавление посещаемости группы A-F-K в предмет криптоварение

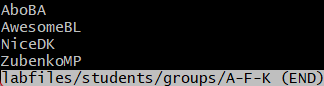
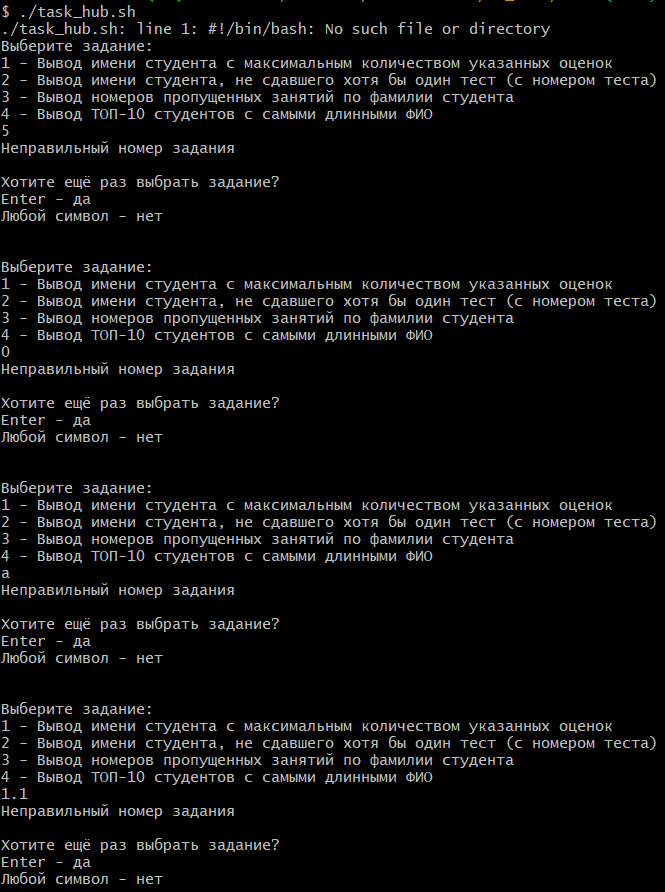


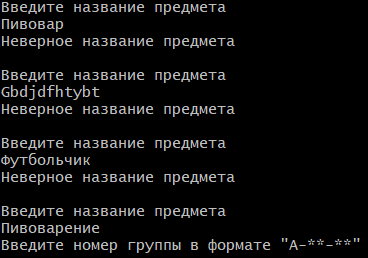
Рисунок 4 – добавление списка группы A-F-K в папку students/groups

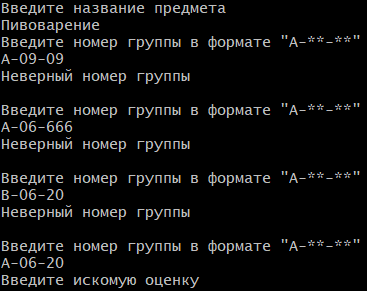
Для тестирования скриптов task1-4.sh были созданы скрипты task1-4\_test.sh. Их код будет приведён в соответствующих подпунктах.

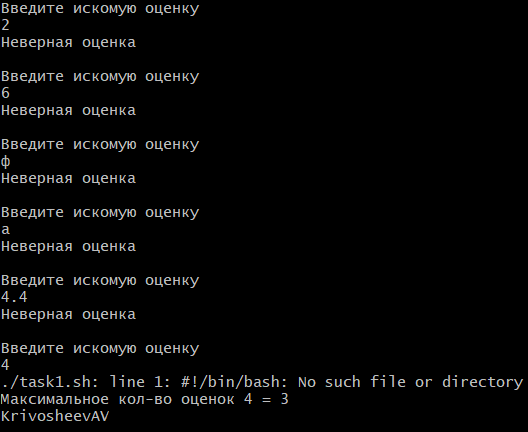
## **task\_hub.sh**

Основное тестирование будет удобнее производить в скриптах-тасках, в хабе проверим обработчик ошибок ввода:

  
Рисунок 5 – Проверка ошибок при выборе задачи

  
Рисунок 6 – Проверка обработки ошибок ввода названия предмета

  
Рисунок 7 – Проверка обработки ошибок ввода номера группы

  
Рисунок 8 – Проверка обработки ошибок ввода оценок

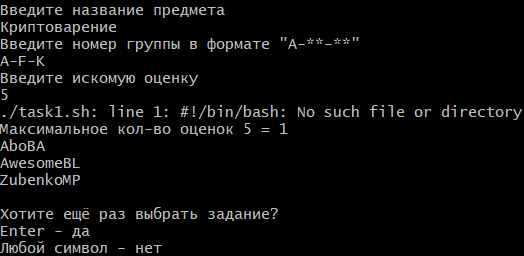


Рисунок 9 – Проверка работы с новым предметом криптоварение

## **task1.sh**

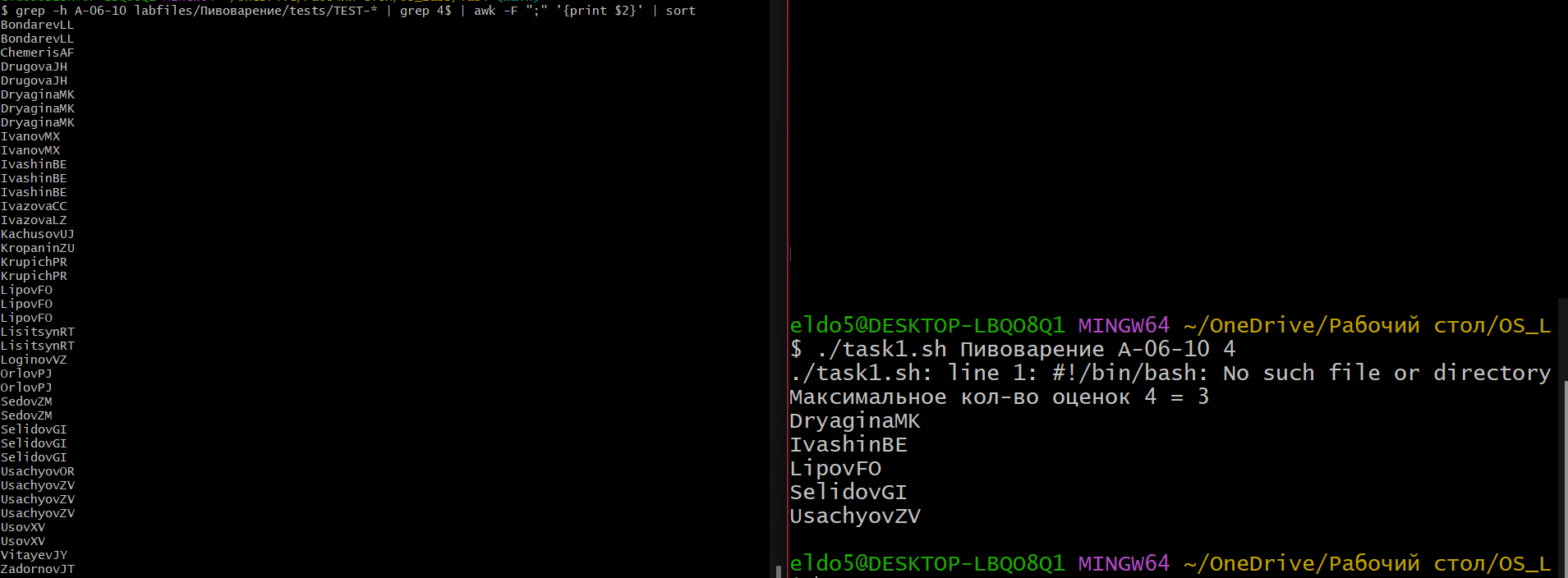


Рисунок 10 – Процесс тестирования скрипта task1.sh

### **task1\_test.sh**

echo --------------------------------------------------------------------------------------------\*task1\_tests

echo ----------------------------------------------\*test1

diff <(bash task1.sh Пивоварение a-06-20 5) <(echo -e "Максимальное кол-во оценок 5 = 3\nDereberiaTM\nSanjiyevMJ") -s

echo ----------------------------------------------\*test2

diff <(bash task1.sh Пивоварение A-06-20 4) <(echo -e "Максимальное кол-во оценок 4 = 3\nKrivosheevAV") -s

echo ----------------------------------------------\*test3

diff <(bash task1.sh Пивоварение A-06-20 3) <(echo -e "Максимальное кол-во оценок 3 = 2\nBespalovML\nDavydovAlD\nDolgyAV\nYankinMI\nZinchenkoSK") -s

echo ----------------------------------------------\*test4

diff <(bash task1.sh Криптозоология A-06-08 3) <(echo -e "Максимальное кол-во оценок 3 = 3\nDubovNO") -s

echo ----------------------------------------------\*test5

diff <(bash task1.sh Криптозоология A-09-20 4) <(echo -e "Максимальное кол-во оценок 4 = 4\nBijoyA") -s

echo ----------------------------------------------\*test6

diff <(bash task1.sh Криптозоология A-09-19 5) <(echo -e "Максимальное кол-во оценок 5 = 3\nKhodeevYI\nOsipovYegY\nRybtsovaAV") -s

echo ----------------------------------------------\*test7

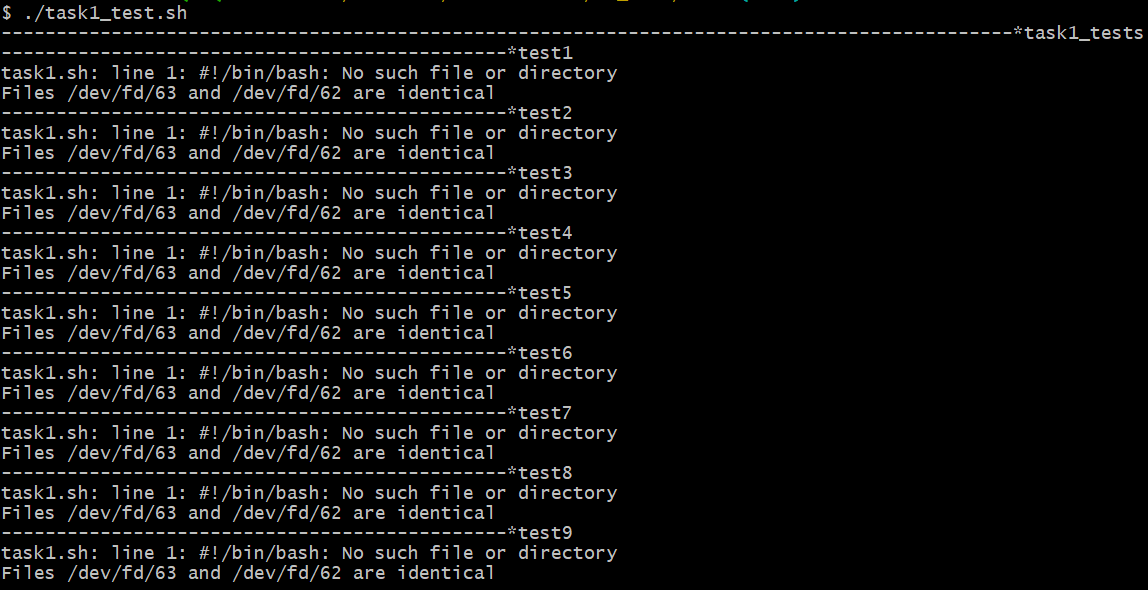
diff <(bash task1.sh Криптоварение A-F-K 5) <(echo -e "Максимальное кол-во оценок 5 = 1\nAboBA\nAwesomeBL\nZubenkoMP") -s

echo ----------------------------------------------\*test8

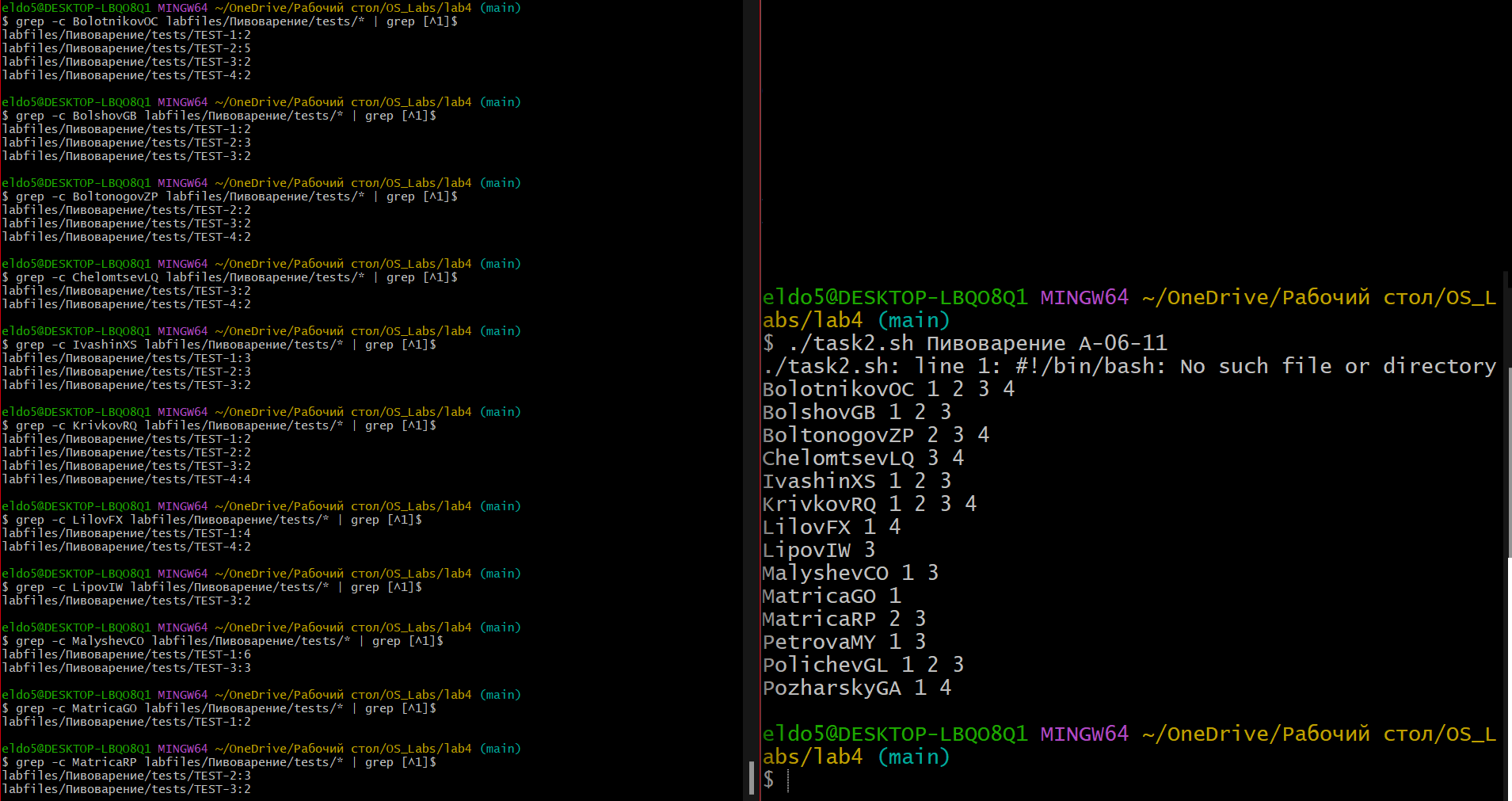
diff <(bash task1.sh Криптоварение A-F-K 4) <(echo -e "Максимальное кол-во оценок 4 = 1\nNiceDK") -s

echo ----------------------------------------------\*test9

diff <(bash task1.sh Криптоварение A-F-K 3) <(echo -e "Нет людей, получивших оценку 3") -s

  
Рисунок 11 – Вывод скрипта task1\_test.sh

## **task2.sh**

  
Рисунок 12 – Процесс тестирования task2.sh

### **task2\_test.sh**

#!/bin/bash

echo --------------------------------------------------------------------------------------------\*task2\_tests

echo ----------------------------------------------\*test1

diff <(bash task2.sh Пивоварение a-06-20) <(echo -e "BespalovML 1 2 3\nChernyshovDM 3\nDavydovAlD 2 4\nDereberiaTM 4\nDolgyAV 1 4\nKrivosheevAV 1 2\nNazhmetdinovMA 1 4\nRybakovArV 1 2\nSanjiyevMJ 2\nTereshinDP 1 2\nTretyakovAlO 4\nYankinMI 2 3\nZhidkikhYS 1 2 4\nZinchenkoSK 4") -s

echo ----------------------------------------------\*test2

diff <(bash task2.sh Пивоварение A-09-17) <(echo -e "FrolovSerAl 2 4\nGromovMN 1 3\nKukinVA 2\nLebedevAlD 4\nMakhalovaOS 2 4\nMarchukON 1\nStudenikinYV 1 2 3\nVlasovaOM 2 3\nZagainovaMD 1 2 4") -s

echo ----------------------------------------------\*test3

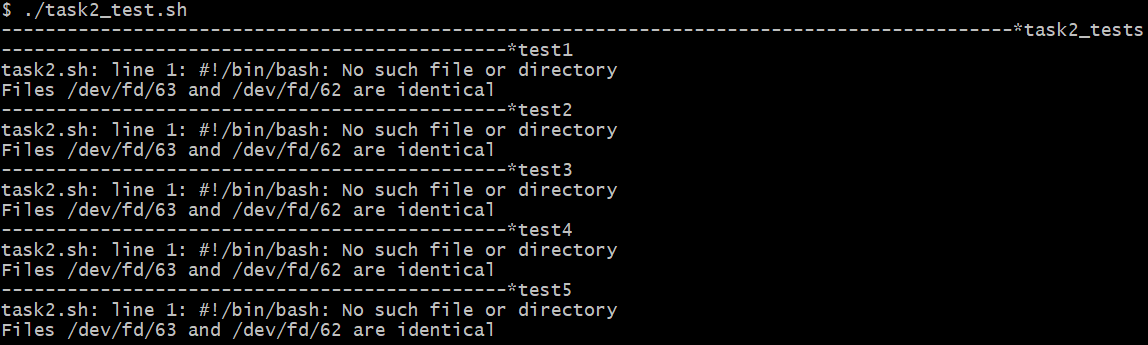
diff <(bash task2.sh Пивоварение A-06-08) <(echo -e "DubovNO 1 4\nLimonovJM 3\nMalakhovaFR 1 3 4\nPodaKS 1 2\nPogrebnovWM 1 2\nShipovUU 1 2 3 4\nZabolotnyDY 3 4") -s

echo ----------------------------------------------\*test4

diff <(bash task2.sh Криптозоология A-06-08) <(echo -e "DubovNO 4\nLidinaSM 2\nLimonovJM 1 2 3 4\nMalakhovaFR 2 3\nPodaKS 1 2 3\nPogrebnovWM 2 3 4\nShipovUU 4\nZabolotnyDY 2 3 4") -s

echo ----------------------------------------------\*test5

diff <(bash task2.sh Криптоварение A-F-K) <(echo -e "Все студенты сдали тесты") -s

  
Рисунок 13 – Вывод скрипта task2\_test.sh

## **task3.sh**

### **task3\_test.sh**

#!/bin/bash

echo --------------------------------------------------------------------------------------------\*task3\_tests

echo ----------------------------------------------\*test1

diff <(bash task3.sh Пивоварение krivo) <(echo -e "KrivosheevAV\nНомера пропущенных занятий\n2, 8, 11, 13, 14, 17") -s

echo ----------------------------------------------\*test2

diff <(bash task3.sh Пивоварение Makarov) <(echo -e "MakarovDanA\nНомера пропущенных занятий\n4") -s

echo ----------------------------------------------\*test3

diff <(bash task3.sh Криптоварение Abo) <(echo -e "AboBA\nНомера пропущенных занятий\n1") -s

echo ----------------------------------------------\*test4

diff <(bash task3.sh Криптоварение Aweso) <(echo -e "AwesomeBL\nНомера пропущенных занятий\n1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26") -s

echo ----------------------------------------------\*test5

diff <(bash task3.sh Криптоварение Nice) <(echo -e "NiceDK\nНет пропущенных занятий") -s

echo ----------------------------------------------\*test6

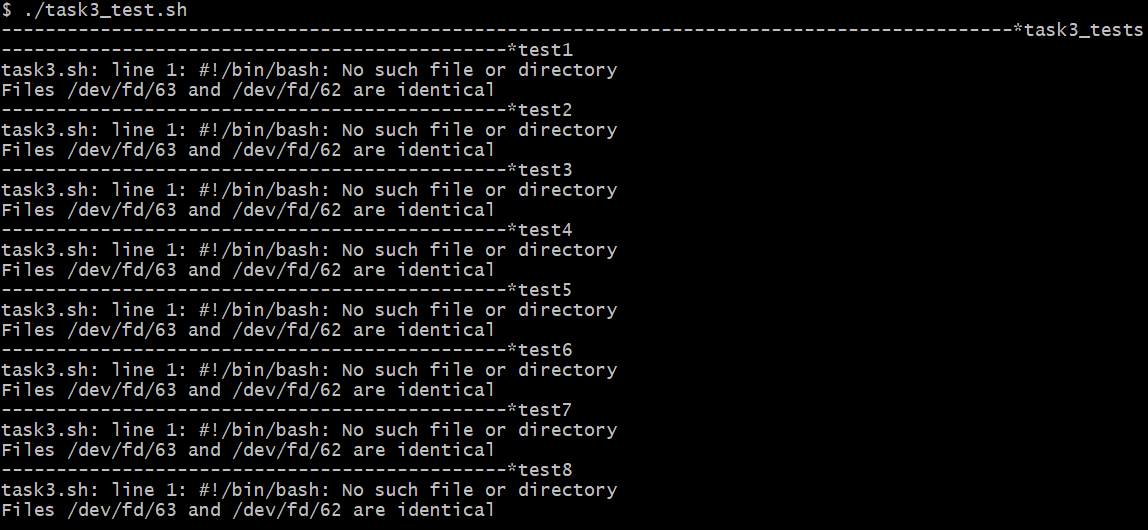
diff <(bash task3.sh Криптоварение Zuben) <(echo -e "ZubenkoMP\nНет пропущенных занятий") -s

echo ----------------------------------------------\*test7

diff <(bash task3.sh Криптозоология Krivo) <(echo -e "KrivosheevAV\nНомера пропущенных занятий\n5, 14, 15, 17, 18") -s

echo ----------------------------------------------\*test8

diff <(bash task3.sh Криптозоология Makarov) <(echo -e "MakarovDanA\nНомера пропущенных занятий\n11, 18") -s

  
Рисунок 14 – Вывод скрипта task3\_test.sh

## **task4.sh**

### **task4\_test.sh**

#!/bin/bash

echo --------------------------------------------------------------------------------------------\*task4\_tests

echo ----------------------------------------------\*test1

diff <(bash task4.sh a-06-20) <(echo -e "1. NazhmetdinovMA\n2. ChernyshovDM\n3. KrivosheevAV\n4. TretyakovAlO\n5. DereberiaTM\n6. ZinchenkoSK\n7. BespalovML\n8. BoldyrevIN\n9. DavydovAlD\n10. RybakovArV") -s

echo ----------------------------------------------\*test2

diff <(bash task4.sh A-09-20) <(echo -e "1. MukhamejanovAI\n2. KharitonovAnA\n3. TikhonovAVad\n4. GrebeniukKS\n5. MikhailovPA\n6. FurtayevIV\n7. GumerovaEF\n8. LabanovaMS\n9. SukhanovVS\n10. AslanovAF") -s

echo ----------------------------------------------\*test3

diff <(bash task4.sh A-F-K) <(echo -e "1. AwesomeBL\n2. ZubenkoMP\n3. NiceDK\n4. AboBA") -s

echo ----------------------------------------------\*test4

diff <(bash task4.sh A-06-08) <(echo -e "1. MalakhovaFR\n2. PogrebnovWM\n3. ZabolotnyDY\n4. LimonovJM\n5. LidinaSM\n6. ShipovUU\n7. DubovNO\n8. PodaKS") -s

echo ----------------------------------------------\*test5

diff <(bash task4.sh) <(echo -e "1. Surovtsev-ButovA\n2. SemichayevskyFF\n3. ChakolyadenisE\n4. CherenchikovJH\n5. MukhamejanovAI\n6. NazhmetdinovMA\n7. PodshivalovaBH\n8. PodshivalovaBZ\n9. PodshivalovaOR\n10. AbramenkoSemV") -s

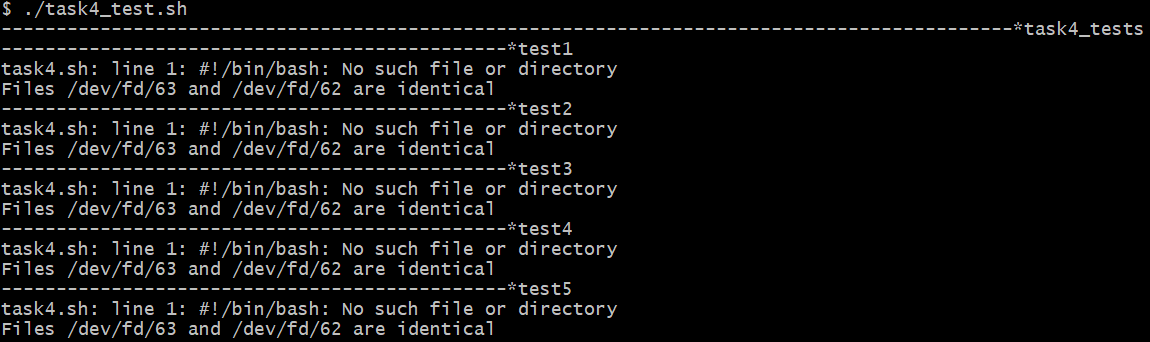


Рисунок 15 – Вывод скрипта task4\_test.sh