

# Отчёт по лабораторной работе №4

Дисциплина: Архитектура компьютера

Студент: Верниковская Екатерина Андреевна

# Содержание

Цель работы	5
Задание	6
Выполнение лабораторной работы	7
Программа Hello world! . . . . .	7
Транслятор NASM . . . . .	9
Расширенный синтаксис командной строки NASM . . . . .	9
Компоновщик LD . . . . .	10
Запуск исполняемого файла . . . . .	10
Задание для самостоятельной работы . . . . .	10
Выводы	16

## Список таблиц

## Список иллюстраций

1	Создание каталога . . . . .	7
2	Переход в каталог . . . . .	7
3	Создание текстового файла . . . . .	8
4	Открытие файла . . . . .	8
5	Ввод программы . . . . .	9
6	Транслирование и проверка . . . . .	9
7	Компиляция и проверка . . . . .	9
8	Выполнение компоновки и проверка 1 . . . . .	10
9	Выполнение компоновки и проверка 2 . . . . .	10
10	Запуск «hello» . . . . .	10
11	Создание нового файла . . . . .	11
12	Открытие нового файла . . . . .	11
13	Изменения в файле . . . . .	12
14	Транслирование . . . . .	12
15	Проверка . . . . .	12
16	Компиляция и проверка . . . . .	12
17	Выполнение компоновки и проверка 1 . . . . .	13
18	Выполнение компоновки и проверка 2 . . . . .	13
19	Запуск «lab4» . . . . .	13
20	Копирование файлов в локальный репозиторий . . . . .	13
21	Проверка 1 . . . . .	14
22	Проверка 2 . . . . .	14
23	Загрузка файлов на github . . . . .	14
24	Проверка в github . . . . .	15

## Цель работы

Освоить процедуры компиляции и сборки программ, написанных на ассемблере NASM.

# Задание

1. Создать каталог для работы с программами на языке ассемблера NASM.
2. Создать текстовый файл с именем «hello.asm» и открыть этот файл с помощью текстового редактора.
3. Ввести в текстовый файл нужный текст.
4. Оттранслировать полученный текст программы «hello.asm» в объектный файл.
5. Выполнить компоновку объектного файла и запустить получившийся исполняемый файл.
6. Создать копию файла «hello.asm» с именем «lab4.asm» и проделать с ним те же действия, что и с файлом «hello.asm».
7. Скопировать файлы «hello.asm» и «lab4.asm» в локальный репозиторий в каталог «~/work/study/2023-2024/“Архитектура компьютера”/arch-rc/labs/lab04/».
8. Загрузить файлы на Github.

# Выполнение лабораторной работы

## Программа Hello world!

Создаём каталог для работы с программами на языке ассемблера NASM с помощью команды ‘mkdir -p’ (рис. [-@fig:001])

```
eavernikovskaya@ubuntu-katerok:~$ mkdir -p ~/work/arch-pc/lab04  
eavernikovskaya@ubuntu-katerok:~$
```

Рис. 1: Создание каталога

Переходим в созданный каталог с помощью ‘cd’ (рис. [-@fig:002])

```
eavernikovskaya@ubuntu-katerok:~$ cd ~/work/arch-pc/lab04  
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 2: Переход в каталог

Создаём текстовый файл с именем «hello.asm» (рис. [-@fig:003])

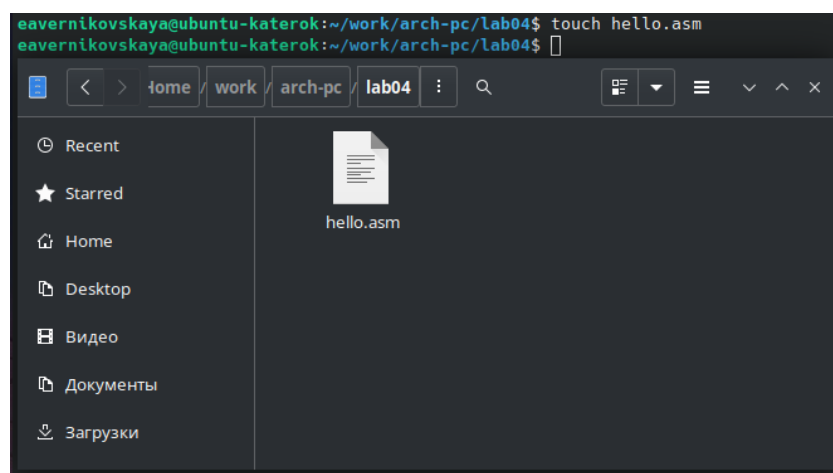


Рис. 3: Создание текстового файла

Открываем файл с помощью текстового редактора «gedit» (рис. [-@fig:004])

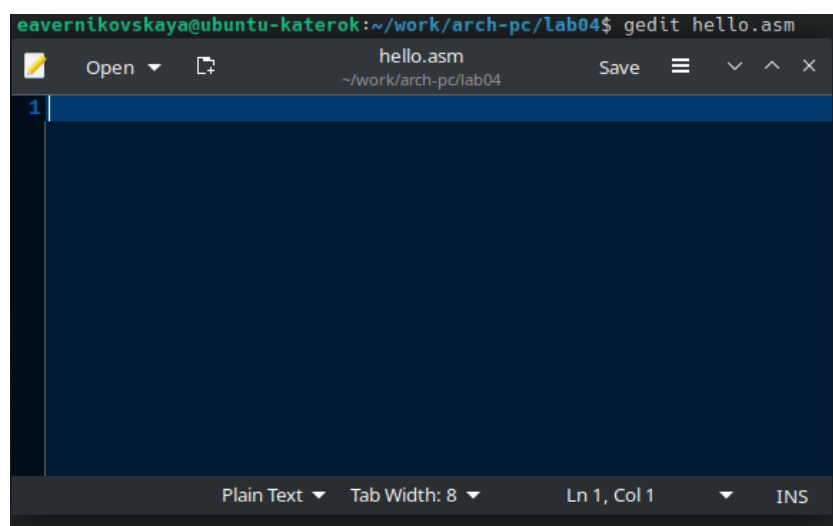
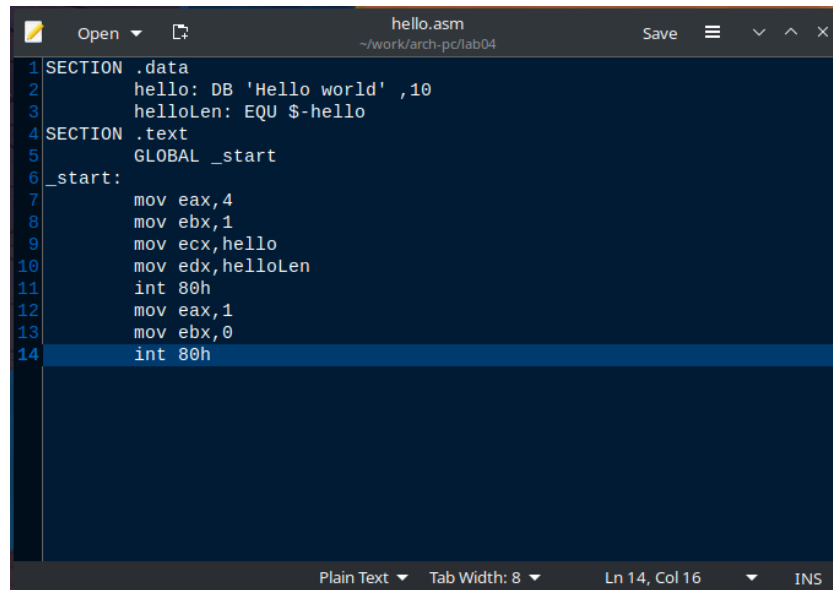


Рис. 4: Открытие файла

Вводим в открытый файл нужный текст (рис. [-@fig:005])



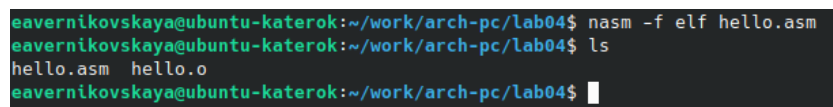


```
1 SECTION .data
2     hello: DB 'Hello world' ,10
3     helloLen: EQU $-hello
4 SECTION .text
5     GLOBAL _start
6 _start:
7     mov eax,4
8     mov ebx,1
9     mov ecx,hello
10    mov edx,helloLen
11    int 80h
12    mov eax,1
13    mov ebx,0
14    int 80h
```

Рис. 5: Ввод программы

## Транслятор NASM

С помощью ‘nasm -f elf’ транслируем текст программы в объектный файл и проверяем с помощью ‘ls’ (рис. [-@fig:006])

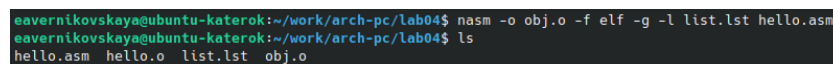


```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ nasm -f elf hello.asm
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 6: Транслирование и проверка

## Расширенный синтаксис командной строки NASM

Компилируем исходный файл «hello.asm» в «obj.o» с помощью ‘nasm -o’ и ‘-f elf -g -l’ и проверяем с помощью ‘ls’ (рис. [-@fig:007])



```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello.asm  hello.o  list.lst  obj.o
```

Рис. 7: Компиляция и проверка

## Компоновщик LD

Передаём объектный файл «hello.o» на обработку компоновщику с помощью ‘ld -m elf\_i386 ... -o’ и проверяем с помощью ‘ls’ (рис. [-@fig:008])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
```

Рис. 8: Выполнение компоновки и проверка 1

Выполняем следующую команду ‘ld -m elf\_i386 obj.o -o main’ (рис. [-@fig:009])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
```

Рис. 9: Выполнение компоновки и проверка 2

Ответ на вопрос: 1) Исполняемый файл будет иметь имя «main» 2) Объектный файл из которого собран этот исполняемый называется «obj.o»

## Запуск исполняемого файла

С помощью ‘./hello’ запускаем исполняемый файл (рис. [-@fig:010])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ./hello
Hello world
```

Рис. 10: Запуск «hello»

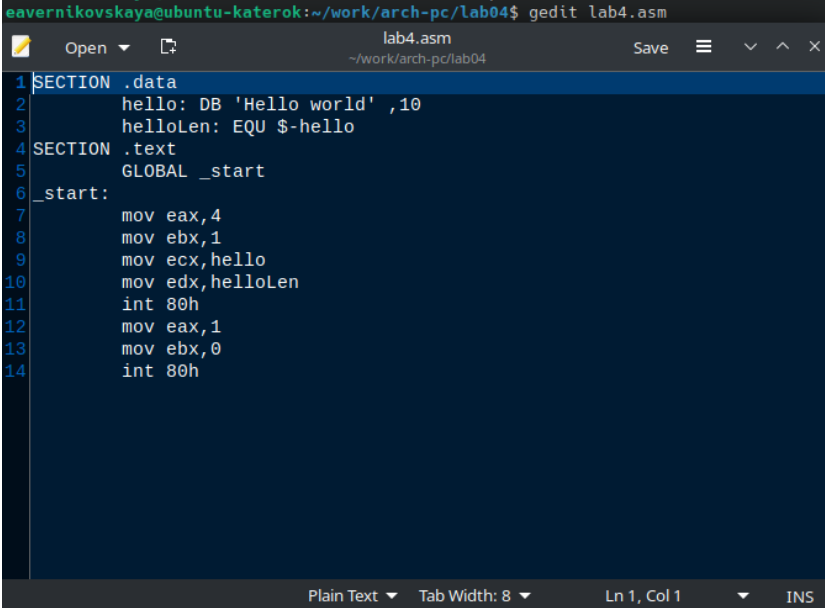
## Задание для самостоятельной работы

С помощью ‘cp’ создаём копию файла «hello.asm» с именем «lab4.asm» и проверяем с помощью ‘ls’ (рис. [-@fig:011])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
```

Рис. 11: Создание нового файла

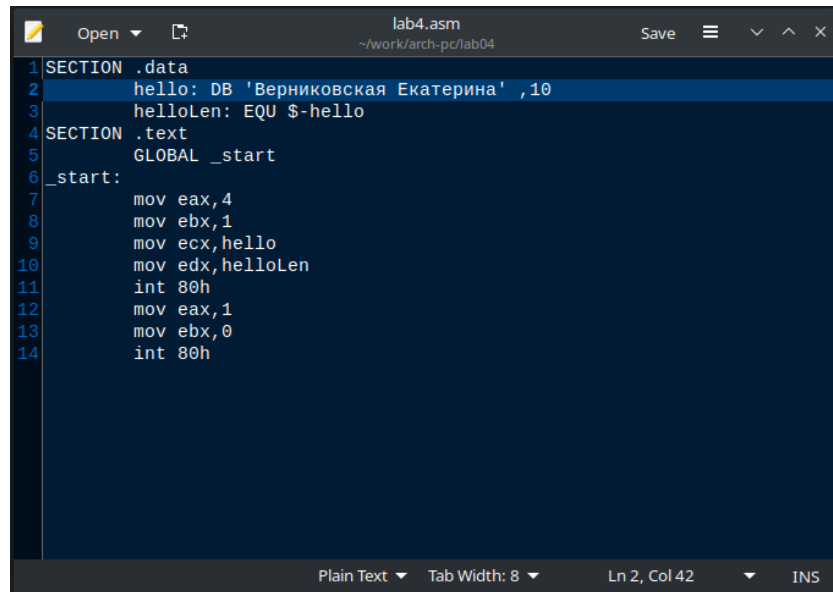
С помощью текстового редактора «gedit» открываем файл «lab4.asm» (рис. [-@fig:012])



```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ gedit lab4.asm
lab4.asm
~/work/arch-pc/lab04
1 SECTION .data
2     hello: DB 'Hello world' ,10
3     helloLen: EQU $-hello
4 SECTION .text
5     GLOBAL _start
6 _start:
7     mov eax,4
8     mov ebx,1
9     mov ecx,hello
10    mov edx,helloLen
11    int 80h
12    mov eax,1
13    mov ebx,0
14    int 80h
Plain Text Tab Width: 8 Ln 1, Col 1 INS
```

Рис. 12: Открытие нового файла

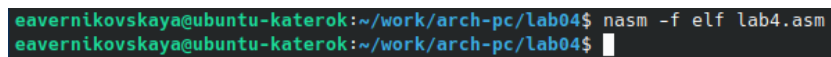
Вносим изменение в текстовый файл так, чтобы на экран выводилась строка с нашим именем и фамилией (рис. [-@fig:013])



```
1 SECTION .data
2 hello: DB 'Верниковская Екатерина' ,10
3 helloLen: EQU $-hello
4 SECTION .text
5 GLOBAL _start
6 _start:
7     mov eax,4
8     mov ebx,1
9     mov ecx,hello
10    mov edx,helloLen
11    int 80h
12    mov eax,1
13    mov ebx,0
14    int 80h
```

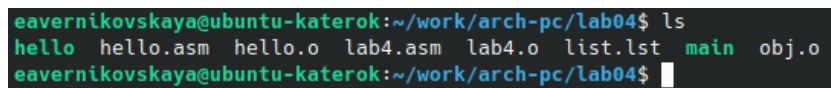
Рис. 13: Изменения в файле

С помощью ‘nasm -f elf’ транслируем текст программы в объектный файл и проверяем с помощью ‘ls’ (рис. [-@fig:014]), (рис. [-@fig:015])



```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

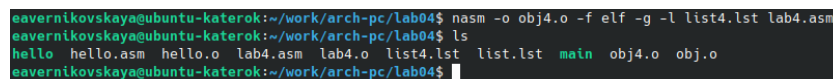
Рис. 14: Транслирование



```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o lab4.asm lab4.o list.lst main obj.o
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 15: Проверка

Компилируем исходный файл «lab4.asm» в «obj4.o» с помощью ‘nasm -o’ и ‘-f elf -g -l’ и проверяем с помощью ‘ls’ (рис. [-@fig:016])



```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ nasm -o obj4.o -f elf -g -l list4.lst lab4.asm
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o lab4.asm lab4.o list4.lst list.lst main obj4.o obj.o
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 16: Компиляция и проверка

Передаём объектный файл «lab4.o» на обработку компоновщику с помощью ‘ld -m elf\_i386 ... -o’ и проверяем с помощью ‘ls’ (рис. [-@fig:017])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list4.lst  list.lst  main  obj4.o  obj.o
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 17: Выполнение компоновки и проверка 1

Передаём объектный файл «obj4.o» на обработку компоновщику с помощью ‘ld -m elf\_i386 ... -o’ и проверяем с помощью ‘ls’ (рис. [-@fig:018])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ld -m elf_i386 obj4.o -o main4
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list4.lst  list.lst  main  main4  obj4.o  obj.o
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 18: Выполнение компоновки и проверка 2

С помощью ‘./lab4’ запускаем исполняемый файл (рис. [-@fig:019])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ ./lab4
Верниковская Екатерина
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 19: Запуск «lab4»

С помощью ‘cp’ копируем файлы «hello.asm» и «lab4.asm» в локальный репозиторий в каталог «~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/» (рис. [-@fig:020])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ cp ~/work/arch-pc/lab04/hello.asm ~/work/study/2023-2024/
"Архитектура компьютера"/arch-pc/labs/lab04/
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ cp ~/work/arch-pc/lab04/lab4.asm ~/work/study/2023-2024/
"Архитектура компьютера"/arch-pc/labs/lab04/
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$
```

Рис. 20: Копирование файлов в локальный репозиторий

Проверяем с помощью ‘ls’ (рис. [-@fig:021]), (рис. [-@fig:022])

```
eavernikovskaya@ubuntu-katerok:~/work/arch-pc/lab04$ cd ~/work/study/2023-2024/"Архитектура компьютера"
/arch-pc/labs/lab04/
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm lab4.asm presentation report
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 21: Проверка 1

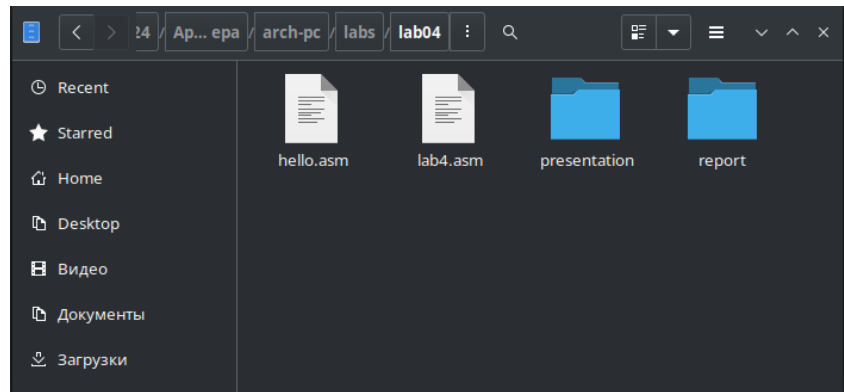


Рис. 22: Проверка 2

Загружаем файлы на github (рис. [-@fig:023])

```
eavernikovskaya@ubuntu-katerok:~$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$ git add .
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$ git commit -am 'add asm'
[master fa4ad1a] add asm
2 files changed, 28 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab4.asm
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 8 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 651 байт | 325.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:Katerok27153/study_2023-2024_arch-pc.git
  87bdd57..fa4ad1a master -> master
eavernikovskaya@ubuntu-katerok:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 23: Загрузка файлов на github

Проверяем. Всё хорошо! (рис. [-@fig:024])

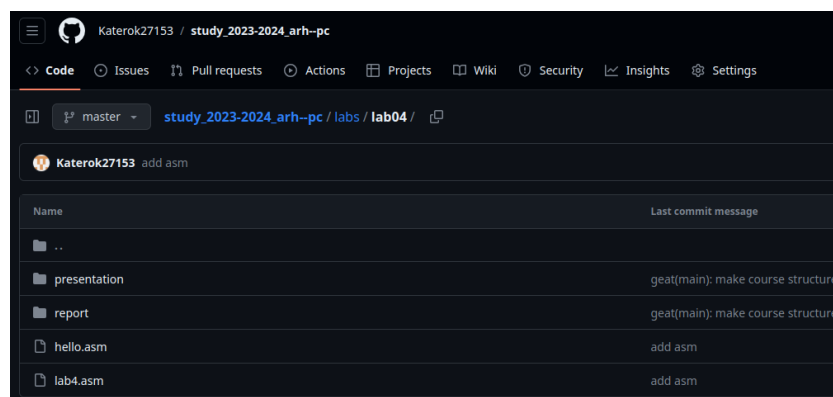


Рис. 24: Проверка в github

## Выводы

В ходе выполнения лабораторной работы мы освоили процедуры компиляции и сборки программ, написанных на ассемблере NASM.