Лабораторная работа №5 Создание и процесс обработки программ на языке ассемблера NASM

Факультет физико-математических и естественных наук Кафедра прикладной информатики и теории вероятностей. Дисциплина: Архитектура ЭВМ

Осокин Геогий Иванович. НММбд-02-22

Содержание

# 1 Цель работы

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

# 2 Выполнение лабораторной работы

## 2.1 Программа “Hello world”

### 2.1.1 Создадим файл hello.asm

Перейдем в директорию lab05

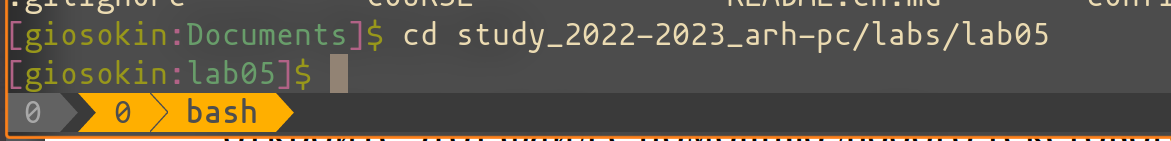


Рис. 1: Переход в директорию lab05

Создадим фалй lab05.asm и откроем его в текстовом редакторе *Emacs*



Рис. 2: Создание lab05.asm

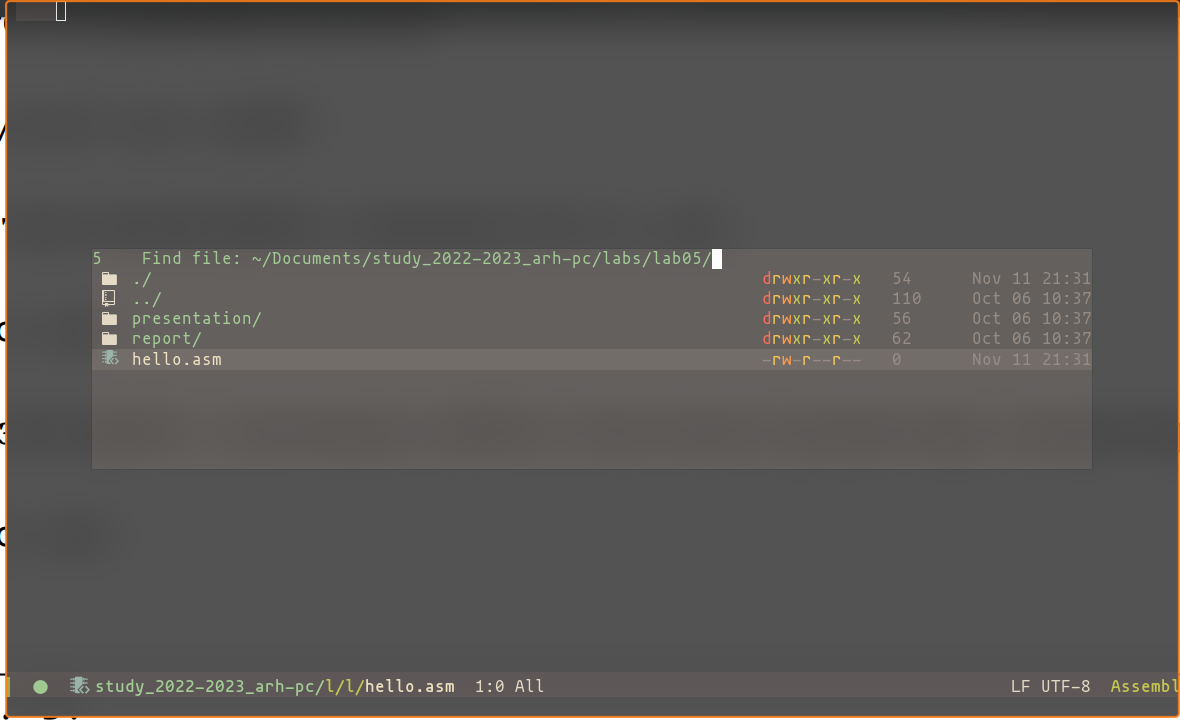


Рис. 3: Открытие файла через тектовый редактор Emacs

Введем исходный текст в файл

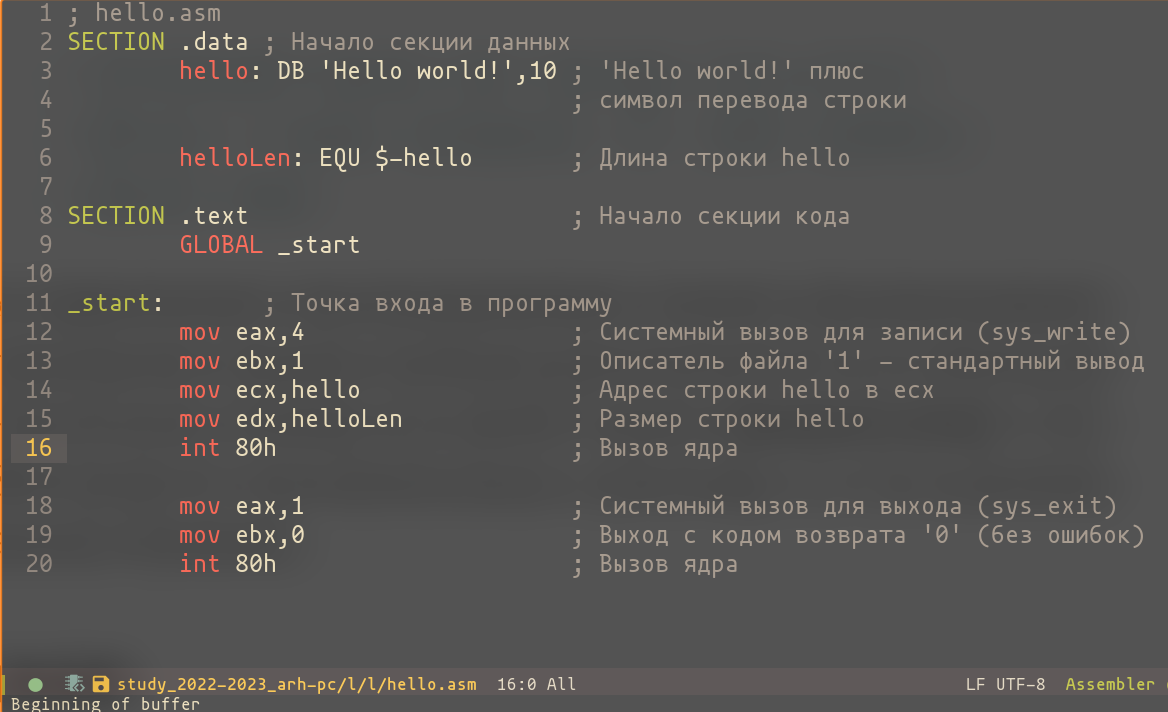


Рис. 4: Код на ассемблере введеный в файл

### 2.1.2 Воспользуемся транслятором NASM

Странслируем исходный код в объектный файл с помощью комманды nasm -f elf hello.asm

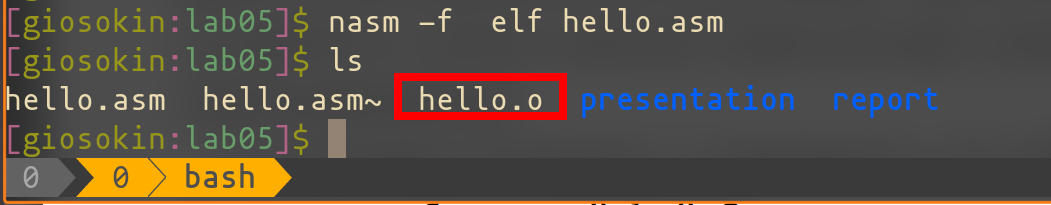


Рис. 5: Трансляция в объектный файл

Создадим объектный файл с другим именем, с помощью опции -o и сгенерируем листинг, с помощью опции -l

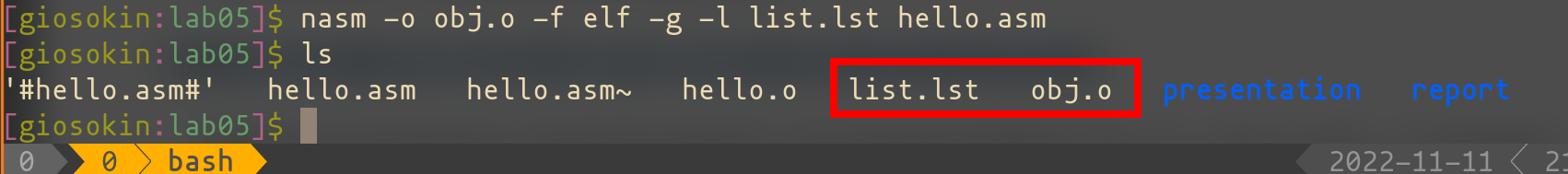


Рис. 6: Создание объектного файла с другим именем и создание листинга

### 2.1.3 Слинкуем obj.o спомощью утилиты ld

Исполним комманду ld -m elf\_i386 hello.o -o hello , что б получить исполняемый файл hello

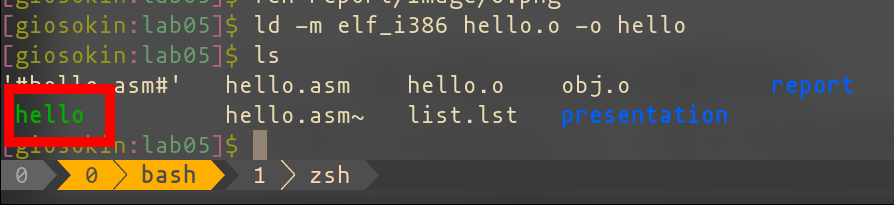


Рис. 7: Линковка с помощью ld

### 2.1.4 Откроем исполняемый файл через дизассемблер

Мы получили исполняемый файл. Из любопытства, откроем его через дизасемблер *Radare2*

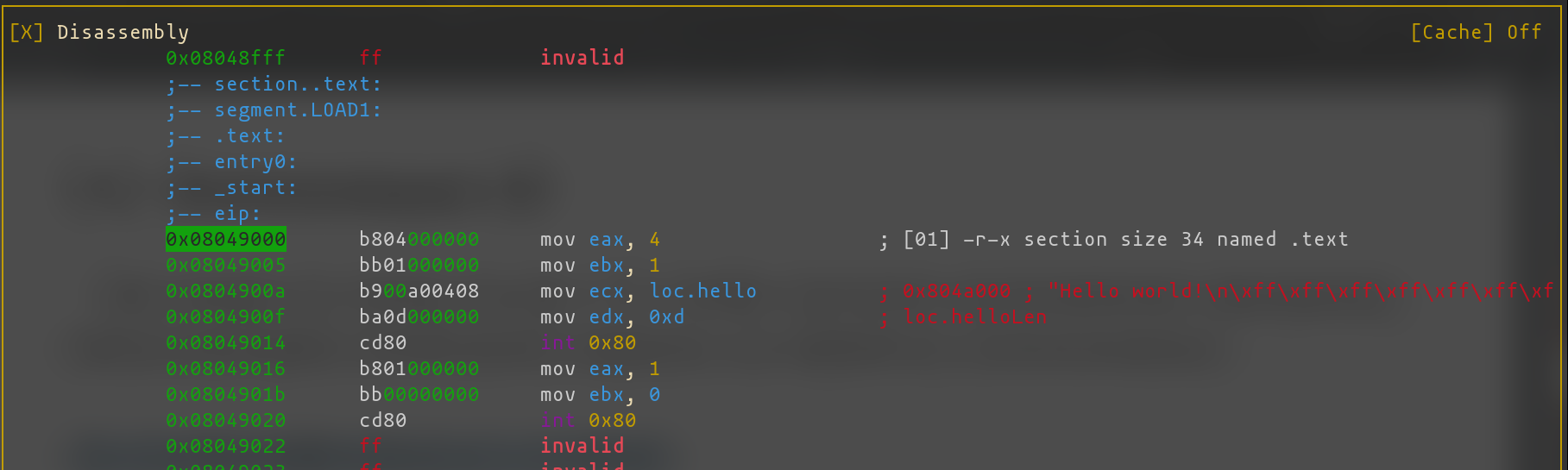


Рис. 8: Просмотр исполняегого файла через дизасемблер radare2

Заметим, что код на ассемблере представленный здесь, очень похож на наш изначальный и даже сохронились названия некоторый “переменных”

### 2.1.5 Слинкуем в файл с другим именем с помощью опции -o

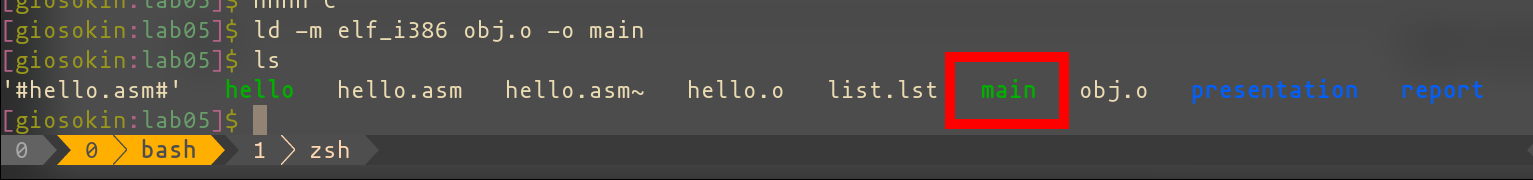


Рис. 9: Создание исполняемого файла с именем main

### 2.1.6 Удалим лишние файлы

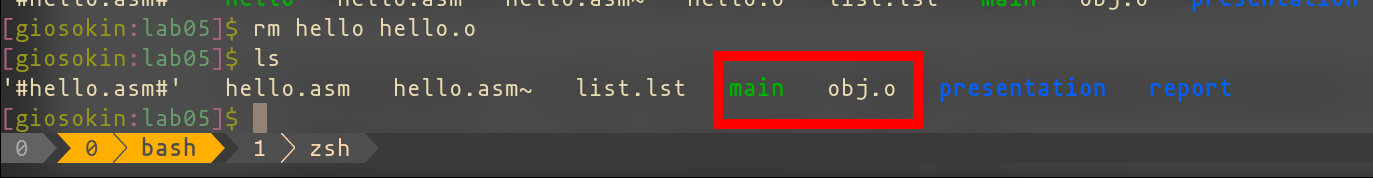


Рис. 10: Удаление лишних файлов

### 2.1.7 Запустим исполняемый файл

Наберем в консоли ./main и увидем вывод.

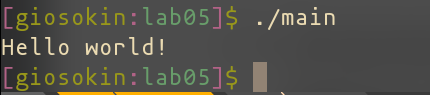


Рис. 11: Исполнение ./main

## 2.2 Задания для самостоятельной работы

### 2.2.1 Создадим файл lab05.asm



Рис. 12: Создание фалйа lab05.asm

Скопируем исходный код из hello.asm и модифицируем его через текстовый редактор *Emacs*

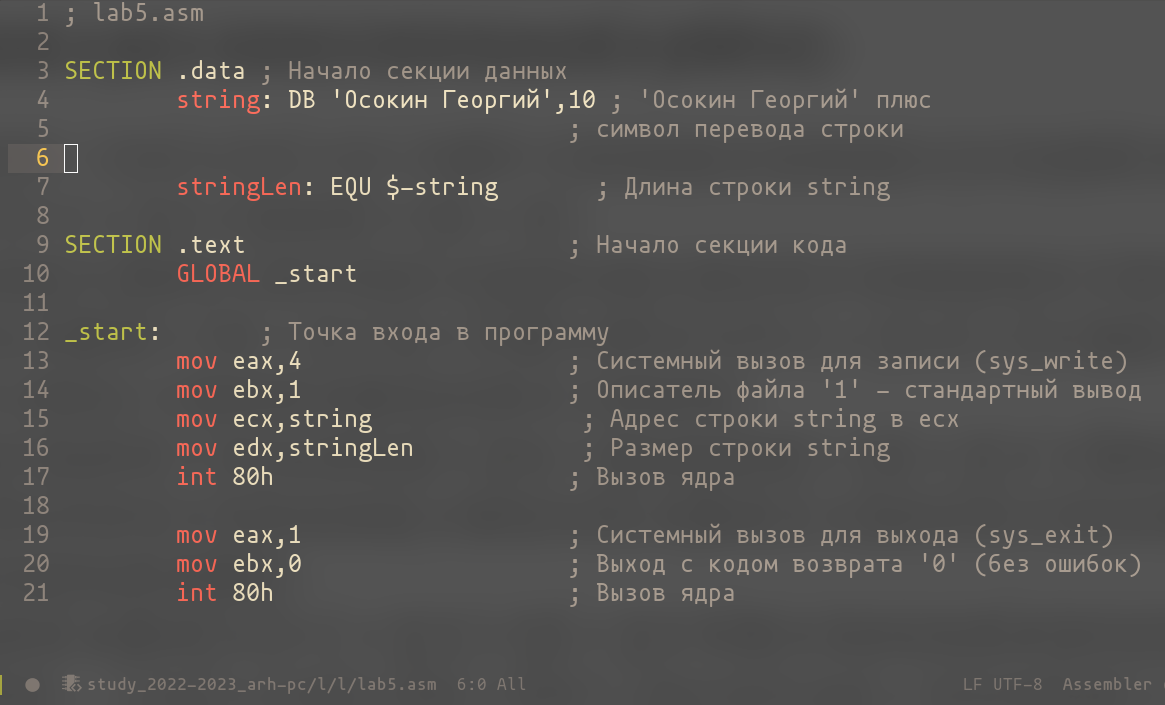


Рис. 13: Содержимое файла lab05.asm

### 2.2.2 Странслируем в объектный файл obj-lab05.o

Воспользуемся транслятором NASM что бы получить объектный файл

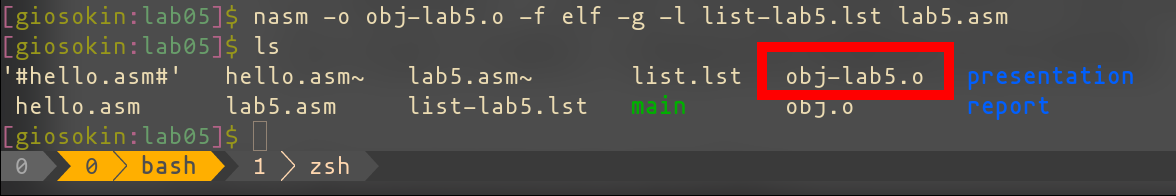


Рис. 14: Трансляцие lab05

### 2.2.3 Создадим исполняемый файл с помощью ld

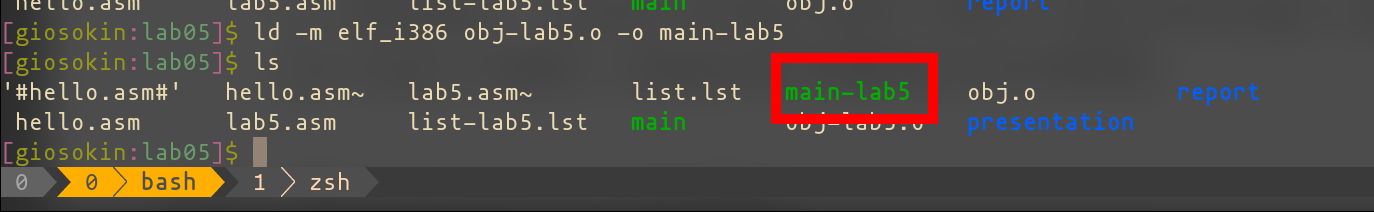


Рис. 15: Создание исполняемого файла main-lab5

### 2.2.4 Запустим ./lab05

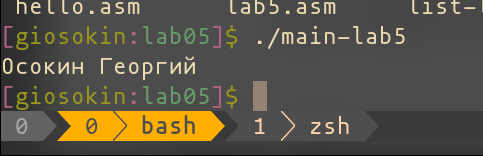


Рис. 16: Исполнение lab05

Как видим, в консоль выводится строка, которую мы задали (фамилия с именем)

## 2.3 Скопируем папку в рабочее пространство и отправим на GitHub

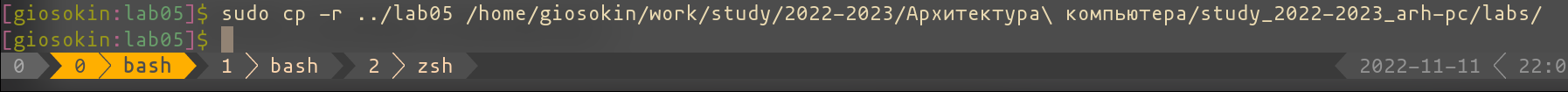


Рис. 17: Копирование в рабочее пространство

Создадим коммит с сообщением **“start lab05”** и запушим.

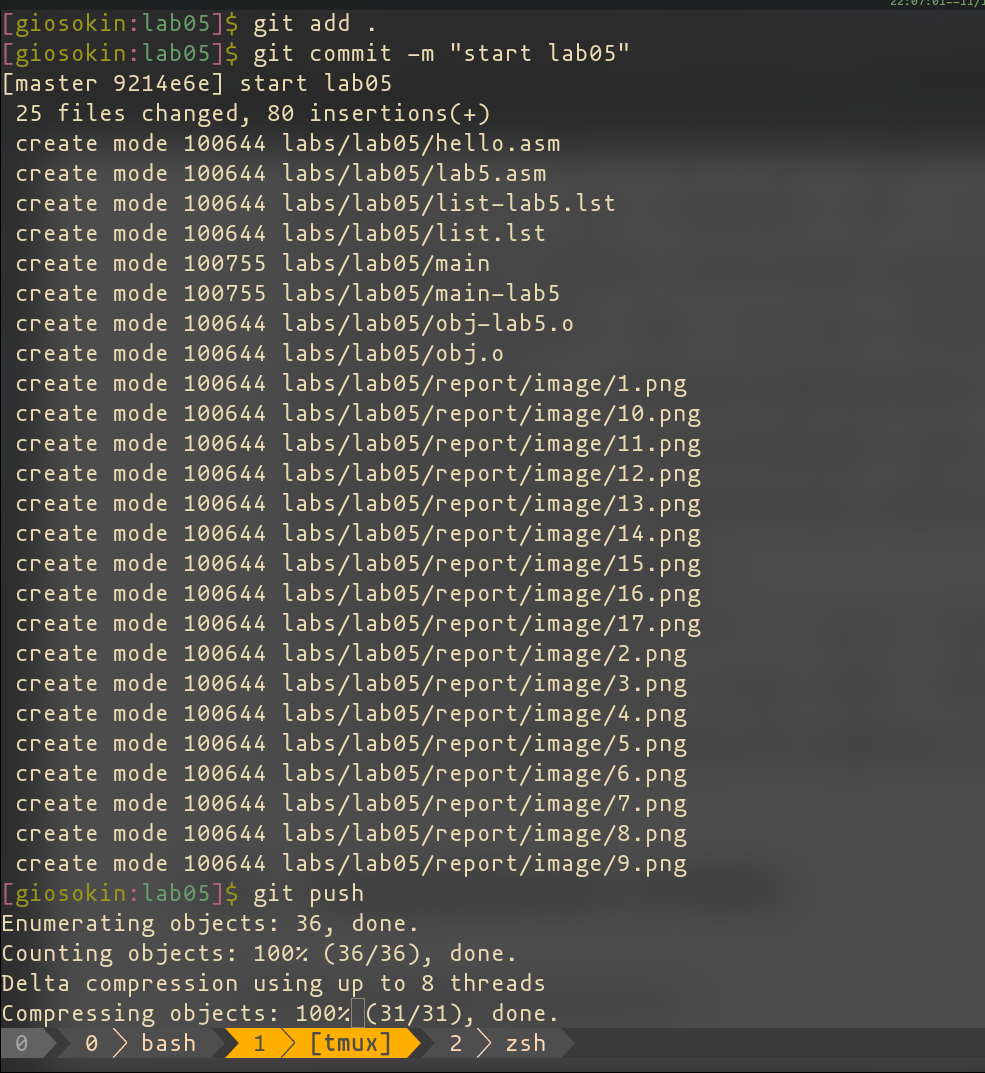


Рис. 18: Создание коммита и отправка файлов на GitHub

# 3 Выводы

В ходе данной лабораторной работы мы научились переводить программы на языке ассемблера NASM в исполняемый файл с помощью трансляции и последуюещей линковки. Мы написали программу на NASM, которая ввыводит в консоль нашу фамилию и имя. А также проверили, что в исполняемом файле в дизасемблированном виде находится очень похожий на наш изначальный код на языке ассемблера.