

Отчёт по лабораторной работе №4

Дисциплина: Архитектура компьютеров

Захаров Кирилл Юрьевич

Цель работы

Цель данной лабораторной работы заключается в изучении и овладении процедурами компиляции и сборки программ, созданных с использованием ассемблера NASM.

Выполнение лабораторной работы

Программа Hello world!

С помощью утилиты `cd` перемещаюсь в каталог, в котором буду работать (рис. 01).

```
kyuzakharov@VirtualBox:~$ cd /home/kyuzakharov/work/study/2023-2024/"Архитектура компьютера"/study_2023-2024_arh-pc/labs/lab04
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Переход в рабочую директорию

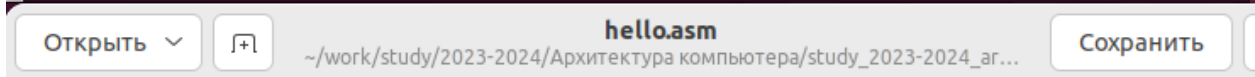
Создаю в текущем каталоге пустой текстовый файл `hello.asm` с помощью утилиты `touch` (рис. 02).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ touch hello.asm
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Создание пустого файла

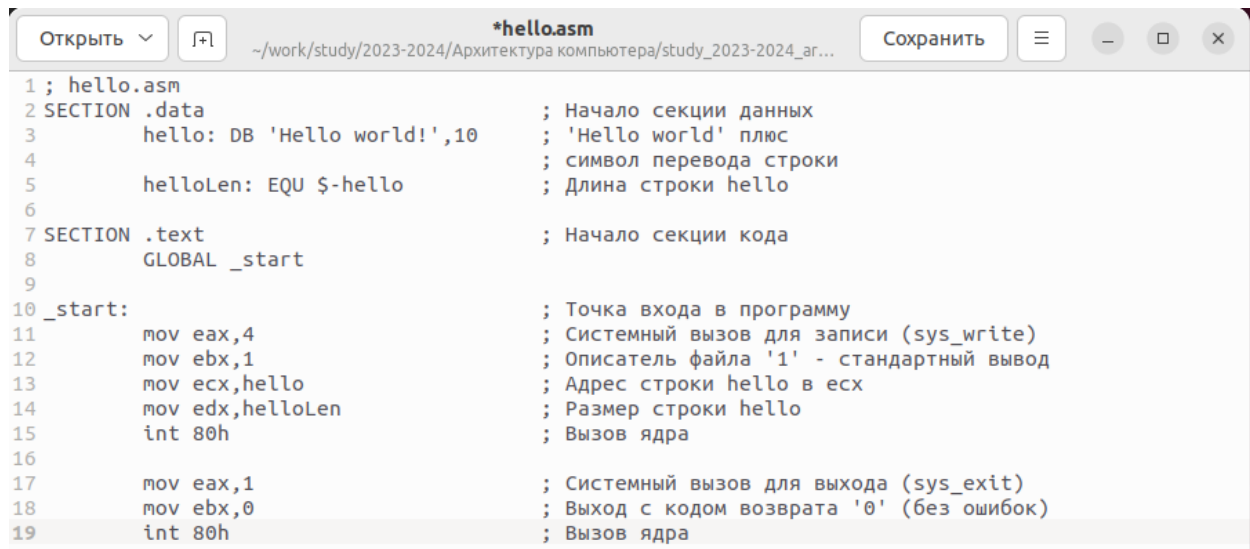
Открываю созданный файл в текстовом редакторе через консоль (рис. 03).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ gedit hello.asm
```



Открытие файла в текстовом редакторе

Заполняю файл, вставляя в него программу для вывода "Hello word!". Так, как ассемблер не является высокоуровневым языком, каждая команда размещается на отдельной строке, так же обращаю внимание на регистр, так как Assembly чувствителен к нему. (рис. 04).

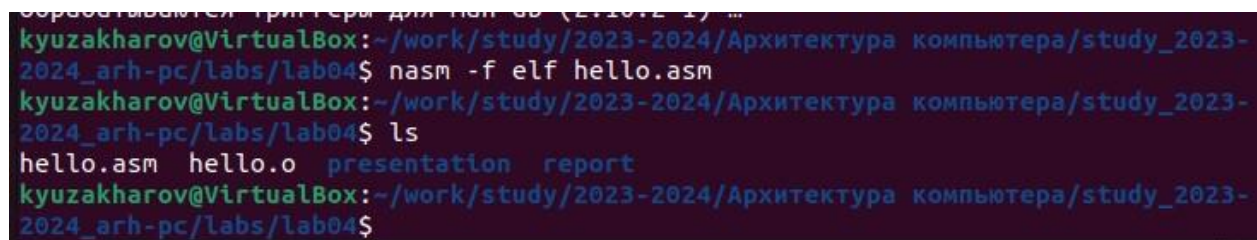


```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3     hello: DB 'Hello world!',10 ; 'Hello world' плюс
4 ; символ перевода строки
5     helloLen: EQU $-hello ; Длина строки hello
6
7 SECTION .text ; Начало секции кода
8     GLOBAL _start
9
10 _start: ; Точка входа в программу
11     mov eax,4 ; Системный вызов для записи (sys_write)
12     mov ebx,1 ; Описатель файла '1' - стандартный вывод
13     mov ecx,hello ; Адрес строки hello в ecx
14     mov edx,helloLen ; Размер строки hello
15     int 80h ; Вызов ядра
16
17     mov eax,1 ; Системный вызов для выхода (sys_exit)
18     mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
19     int 80h ; Вызов ядра
```

Заполнение файла

Транслятор NASM

Превращаю текст программы для вывода “Hello world!” в объектный код с помощью транслятора NASM, используя команду `nasm -f elf hello.asm`, ключ `-f` указывает транслятору `nasm`, что требуется создать бинарный файл в формате ELF (рис. 05). Далее проверяю правильность выполнения команды с помощью утилиты `ls`: действительно, создан файл “hello.o”.

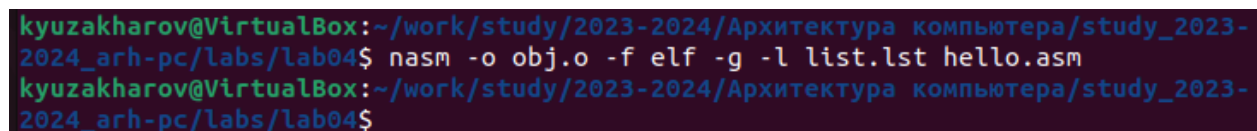


```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ nasm -f elf hello.asm
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello.asm hello.o presentation report
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Компиляция текста программы

Расширенный синтаксис командной строки NASM

Ввожу команду, которая скомпилирует файл `hello.asm` в файл `obj.o`, используя ключ `-o` который задает имя объектному файлу, так же в файл будут включены символы для отладки (ключ `-g`), с помощью ключа `-l` будет создан файл листинга `list.lst` (рис. 06). Далее проверяю с помощью утилиты `ls` правильность выполнения команды (рис. 07).



```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Компиляция текста программы

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello.asm hello.o list.lst obj.o presentation report
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Проверка с помощью команды ls

Компоновщик LD

Передаю объектный файл hello.o на обработку компоновщику LD, чтобы получить исполняемый файл hello (рис. 08). Ключ -o задает имя создаваемого исполняемого файла. Далее проверяю с помощью утилиты ls правильность выполнения команды (рис. 09).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ld -m elf_i386 hello.o -o hello
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Передача объектного файла на обработку компоновщику

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello hello.asm hello.o list.lst obj.o presentation report
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Проверка с помощью команды ls

Выполняю следующую команду (рис. 10). Исполняемый файл будет иметь имя main, т.к. после ключа -o было задано значение main. Объектный файл, из которого собран этот исполняемый файл, имеет имя obj.o

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ld -m elf_i386 obj.o -o main
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Передача объектного файла на обработку компоновщику

Запуск исполняемого файла

Запускаю на выполнение созданный исполняемый файл hello (рис. 11).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ./hello
Hello world!
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Запуск исполняемого файла

Задание для самостоятельной работы.

С помощью утилиты cp создаю в текущем каталоге копию файла hello.asm с именем lab4.asm (рис. 12).

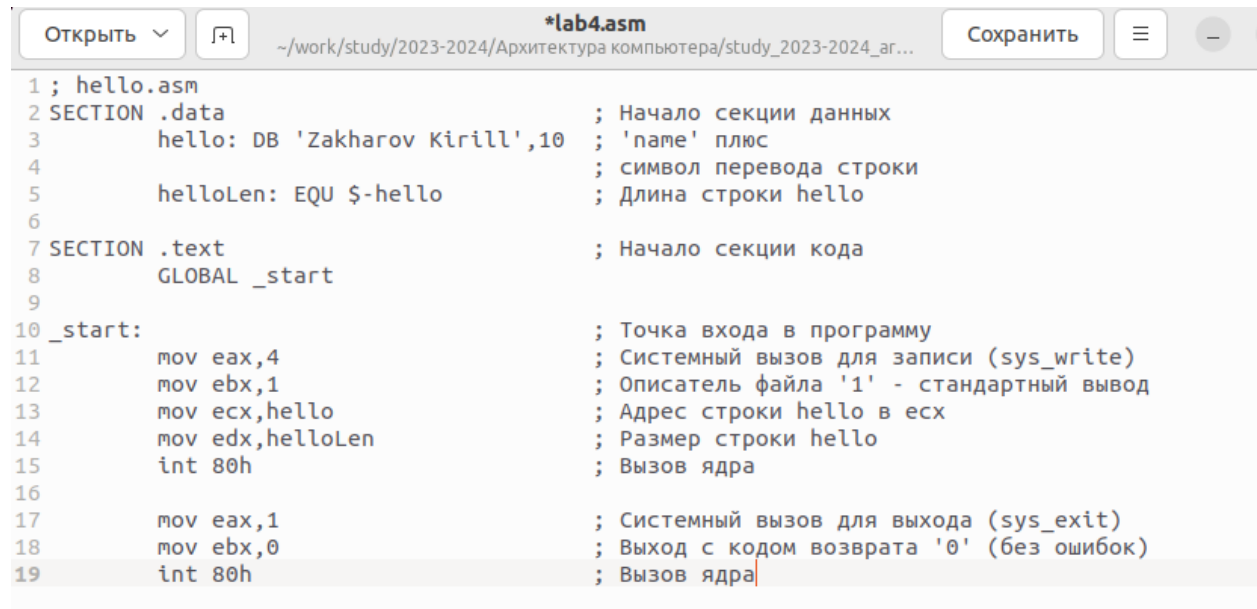
```

kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ cp hello.asm lab4.asm
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello      hello.o  list.lst  obj.o      report
hello.asm  lab4.asm  main      presentation
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$

```

Создание копии файла

С помощью текстового редактора открываю файл lab4.asm и вношу изменения в программу так, чтобы она выводила мои имя и фамилию. (рис. 13).



```

Открыть ▾  [icon]  *lab4.asm  ~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_ar...  Сохранить  ≡  -
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3     hello: DB 'Zakharov Kirill',10 ; 'name' плюс
4 ; символ перевода строки
5     helloLen: EQU $-hello ; Длина строки hello
6
7 SECTION .text ; Начало секции кода
8     GLOBAL _start
9
10 _start: ; Точка входа в программу
11     mov eax,4 ; Системный вызов для записи (sys_write)
12     mov ebx,1 ; Описатель файла '1' - стандартный вывод
13     mov ecx,hello ; Адрес строки hello в ecx
14     mov edx,helloLen ; Размер строки hello
15     int 80h ; Вызов ядра
16
17     mov eax,1 ; Системный вызов для выхода (sys_exit)
18     mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
19     int 80h ; Вызов ядра

```

Изменение программы

Компилирую текст программы в объектный файл (рис. 14). Проверяю с помощью утилиты ls, что файл lab4.o создан.

```

kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ nasm -f elf lab4.asm
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello      hello.o  lab4.o    main      presentation
hello.asm  lab4.asm  list.lst  obj.o      report
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$

```

Компиляция текста программы

Передаю объектный файл lab4.o на обработку компоновщику LD, чтобы получить исполняемый файл lab4 (рис. 15).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ld -m elf_i386 lab4.o -o lab4
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ls
hello      hello.o  lab4.asm  list.lst  obj.o      report
hello.asm  lab4     lab4.o    main      presentation
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Передача объектного файла на обработку компоновщику

Запускаю исполняемый файл lab4, на экран действительно выводятся мои имя и фамилия (рис. 16).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ ./lab4
Zakharov Kirill
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Запуск исполняемого файла

Отправляю файлы на сервер с помощью команды git push (рис. 17).

```
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$ git push -f
Перечисление объектов: 82, готово.
Подсчет объектов: 100% (82/82), готово.
При сжатии изменений используется до 7 потоков
Сжатие объектов: 100% (64/64), готово.
Запись объектов: 100% (82/82), 3.42 МиБ | 3.20 МиБ/с, готово.
Всего 82 (изменений 14), повторно использовано 74 (изменений 11), повторно испол
зовано пакетов 0
remote: Resolving deltas: 100% (14/14), done.
To github.com:kyuzakharov/study_2023-2024_arh-pc.git
+ c7f817c...41a5490 master -> master (forced update)
kyuzakharov@VirtualBox:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab04$
```

Отправка файлов

Выводы

В результате выполнения данной лабораторной работы мы овладели процедурами компиляции и сборки программ, написанных на ассемблере NASM. Эти навыки позволят нам эффективно работать с программами на ассемблере и улучшить навыки низкоуровневого программирования. Мы также научились использовать инструменты для компиляции и сборки программ, что важно для разработки программного обеспечения.