

Лабораторная работа номер 5

Сафин Андрей Алексеевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Самостоятельная работа	9
5	Выводы	12

Список иллюстраций

3.1	Создание директории и файла	7
3.2	Запись программы в файл	7
3.3	Транслирование файла hello.asm	8
3.4	Компоновка и исполнение программы	8
4.1	Процесс создания файла с исполняемой программой	9
4.2	Изменения, внесенные в текст программы	10
4.3	Сохранение результатов на github	11

Список таблиц

1 Цель работы

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

2 Задание

Создать несколько файлов расширения `.pas`, транслировать их в объектный код, скомпоновать их в исполняемую программу и исполнить одну из них.

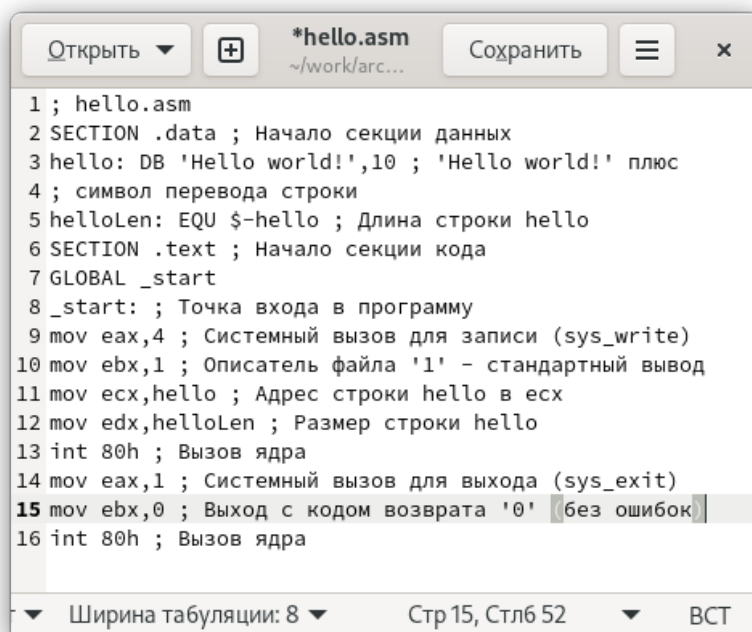
3 Выполнение лабораторной работы

Создана директория `~/work/arch-pc/lab05` и файл `hello.asm`. Файл открыт с помощью текстового редактора (рис. 3.1).

```
[aasafin@fedora ~]$ mkdir -p ~/work/arch-pc/lab05
[aasafin@fedora ~]$ cd ~/work/arch-pc/lab05
[aasafin@fedora lab05]$ touch hello.asm
[aasafin@fedora lab05]$ gedit hello.asm
```

Рис. 3.1: Создание директории и файла

В файл записан код программы, выводящей надпись “Hello world!” (рис. 3.2).



The screenshot shows a text editor window titled `*hello.asm` with the path `~/work/arch...`. The window contains assembly code for a program that prints "Hello world!". The code is as follows:

```
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 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для записи (sys_write)
10 mov ebx,1 ; Описатель файла '1' - стандартный вывод
11 mov ecx,hello ; Адрес строки hello в ecx
12 mov edx,helloLen ; Размер строки hello
13 int 80h ; Вызов ядра
14 mov eax,1 ; Системный вызов для выхода (sys_exit)
15 mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
16 int 80h ; Вызов ядра
```

At the bottom of the window, the status bar shows: `Ширина табуляции: 8`, `Стр 15, Стлб 52`, and `ВСТ`.

Рис. 3.2: Запись программы в файл

Программа транслирована в файл hello.o, а затем в файл obj.o с приложенным файлом листинга (рис. 3.3).

```
[aasafin@fedora lab05]$ nasm -f elf hello.asm
[aasafin@fedora lab05]$ ls
hello.asm  hello.o
[aasafin@fedora lab05]$ nasm -o obj.o -f elf -g -l list.lst h
ello.asm
[aasafin@fedora lab05]$ ls
hello.asm  hello.o  list.lst  obj.o
```

Рис. 3.3: Транслирование файла hello.asm

Файл hello.o скомпонован в файл с тем же названием без расширения, а obj.o - в файл с названием main. Затем выполнена программа hello (рис. 3.4).

```
[aasafin@fedora lab05]$ ld -m elf_i386 hello.o -o hello
[aasafin@fedora lab05]$ ls
hello  hello.asm  hello.o  list.lst  obj.o
[aasafin@fedora lab05]$ ld -m elf_i386 obj.o -o main
[aasafin@fedora lab05]$ ls
hello  hello.asm  hello.o  list.lst  main  obj.o
[aasafin@fedora lab05]$ ./hello
Hello world!
[aasafin@fedora lab05]$
```

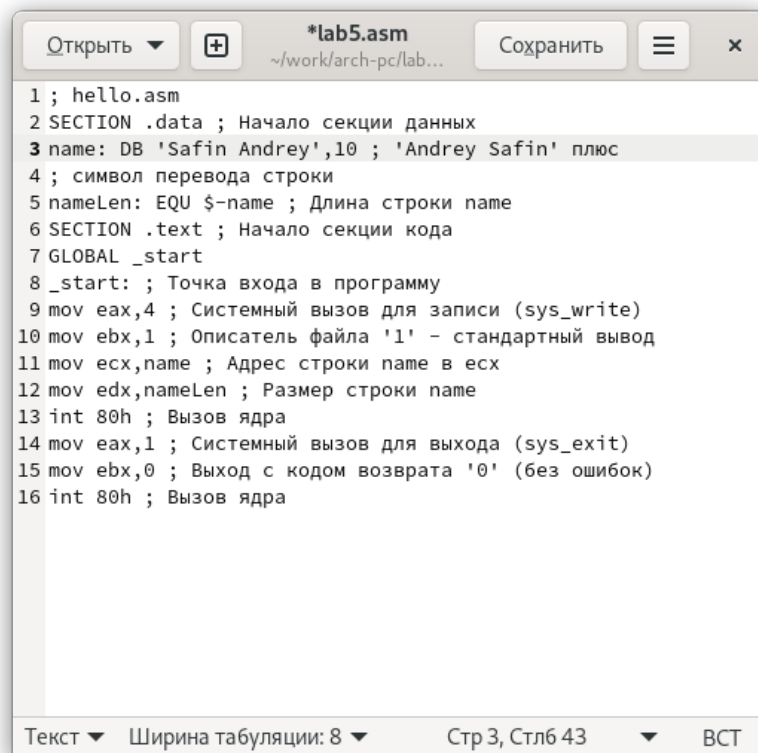
Рис. 3.4: Компоновка и исполнение программы

4 Самостоятельная работа

Продублирован файл `hello.asm` со сменой названия на `lab05.asm` (рис. 4.1). Программа в файле изменена так, чтоб выводилась строка с именем (рис. 4.2). Файл транслирован а затем скомпонован в программу `lab5`. Программа выполнена (рис. 4.1).

```
[aasafin@fedora lab05]$ cp hello.asm lab5.asm
[aasafin@fedora lab05]$ gedit lab5.asm
[aasafin@fedora lab05]$ nasm -f elf lab5.asm
[aasafin@fedora lab05]$ ls
hello.o lab5.o main
hello.asm lab5.asm list.lst obj.o
[aasafin@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
[aasafin@fedora lab05]$ ls
hello.o lab5.asm list.lst obj.o
hello.asm lab5 lab5.o main
[aasafin@fedora lab05]$ ./lab5
Safin Andrey
[aasafin@fedora lab05]$
```

Рис. 4.1: Процесс создания файла с исполняемой программой

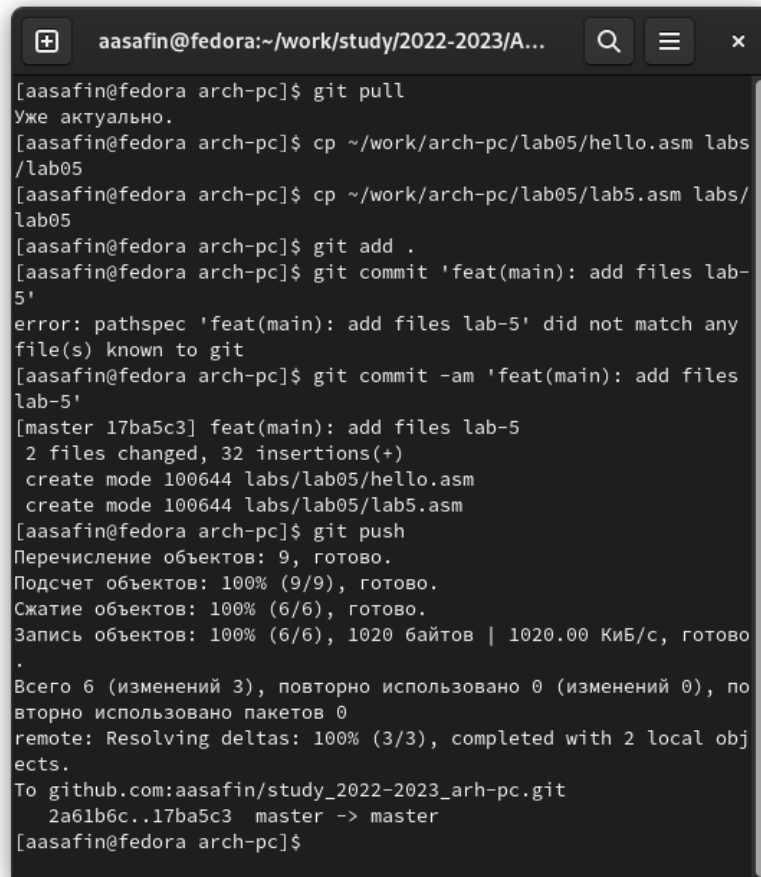


```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 name: DB 'Safin Andrej',10 ; 'Andrej Safin' плюс
4 ; символ перевода строки
5 nameLen: EQU $-name ; Длина строки name
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для записи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

Текст ▾ Ширина табуляции: 8 ▾ Стр 3, Стлб 43 ▾ ВСТ

Рис. 4.2: Изменения, внесенные в текст программы

Затем файлы `hello.asm` и `lab5.asm` скопированы в соответствующую директорию каталога курса. Изменения выгружены на github.



```
[aasafin@fedora arch-pc]$ git pull
Уже актуально.
[aasafin@fedora arch-pc]$ cp ~/work/arch-pc/lab05/hello.asm labs/
lab05
[aasafin@fedora arch-pc]$ cp ~/work/arch-pc/lab05/lab5.asm labs/
lab05
[aasafin@fedora arch-pc]$ git add .
[aasafin@fedora arch-pc]$ git commit 'feat(main): add files lab-
5'
error: pathspec 'feat(main): add files lab-5' did not match any
file(s) known to git
[aasafin@fedora arch-pc]$ git commit -am 'feat(main): add files
lab-5'
[master 17ba5c3] feat(main): add files lab-5
 2 files changed, 32 insertions(+)
 create mode 100644 labs/lab05/hello.asm
 create mode 100644 labs/lab05/lab5.asm
[aasafin@fedora arch-pc]$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 1020 байтов | 1020.00 КиБ/с, готово
.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), по
вторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local obj
ects.
To github.com:aasafin/study_2022-2023_arh-pc.git
 2a61b6c..17ba5c3 master -> master
[aasafin@fedora arch-pc]$
```

Рис. 4.3: Сохранение результатов на github

5 Выводы

Процесс сборки компиляции программ на `pasm` полностью пройден и повторен в рамках самостоятельной работы.