

Лаблораторная работа №5

Создание и процесс обработки программ на языке ассемблера NASM

Кучеренко София

Содержание

| | | |
|---|--------------------------------|---|
| 1 | Цель работы | 5 |
| 2 | Выполнение лабораторной работы | 6 |
| 3 | Самостоятельная работа | 8 |
| 4 | Выводы | 9 |

Список иллюстраций

| | | |
|-----|--|---|
| 2.1 | Создание файла | 6 |
| 2.2 | Текст программы | 6 |
| 2.3 | Создание объектного файла | 7 |
| 2.4 | Компановка | 7 |
| 2.5 | Результат выполнения программы | 7 |
| 3.1 | Создание копии | 8 |
| 3.2 | Измененный текст программы | 8 |
| 3.3 | Результат выполнения программы | 8 |

Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

1. Создадим текстовый файл “hello.asm” и введем в него текст программы:

```
PS C:\Users\user\Desktop\git\study_2022-2023_arh-pc\labs\lab05> touch hello.asm
```

Рис. 2.1: Создание файла

```
> lab05 > ASM hello.asm
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)
9 mov ebx,1 ; Описатель файла '1' - стандартный вывод
1 mov ecx,hello ; Адрес строки hello в ecx
2 mov edx,helloLen ; Размер строки hello
3 int 80h ; Вызов ядра
4 mov eax,1 ; Системный вызов для выхода (sys_exit)
5 mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
6 int 80h ; Вызов ядра
```

Рис. 2.2: Текст программы

2. Создадим объектный файл и исполняемую программу:

```
● @sshkiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ nasm -f elf hello.asm  
● @sshkiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ nasm -o obj.o -f elf -g -l list.lst hello.asm
```

Рис. 2.3: Создание объектного файла

```
● @sshkiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ ld -m elf_i386 hello.o -o hello
```

Рис. 2.4: Компиляция

3. Запустим исполняемый файл:

```
● @sshkiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ ./hello  
Hello world!
```

Рис. 2.5: Результат выполнения программы

3 Самостоятельная работа

1. Создадим копию файла hello.asm с именем lab5.asm:

```
@sshiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ cp hello.asm lab5.asm
```

Рис. 3.1: Создание копии

2. Изменим текст программы, чтобы выводилась другая строка:

```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3     hello: DB 'Кучеренко София',10 ;
4 ; символ перевода строки
5     helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7     GLOBAL _start
```

Рис. 3.2: Измененный текст программы

3. Создадим объектный файл, выполним его компоновку и запустим:

```
@sshiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ cp hello.asm lab5.asm
@sshiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ nasm -f elf lab5.asm
@sshiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ ld -m elf_i386 lab5.o -o lab5
@sshiperr →/workspaces/study_2022-2023_arh-pc/labs/lab05 (master X) $ ./lab5
Кучеренко София
```

Рис. 3.3: Результат выполнения программы

4 Выводы

Я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.