

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

направление: Компьютерные и информационные науки

Лабораторная работа №2

дисциплина: Архитектура компьютеров и операционные системы

студент: Гробман Александр Евгеньевич

Группа: НКАбд-02-23

Цель работы

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

Задание

1. В каталоге `~/work/arch-pc/lab04` с помощью команды `cp` создайте копию файла `hello.asm` с именем `lab4.asm`
2. С помощью любого текстового редактора внесите изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем.
3. Оттранслируйте полученный текст программы `lab4.asm` в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.
4. Скопируйте файлы `hello.asm` и `lab4.asm` в Ваш локальный репозиторий в каталог `~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04/`. Загрузите файлы на Github.
- 5.

Выполнение лабораторной работы

1. **Создание программы 'hello world'**

Выбираем директорию и создаём файл через команду `touch`

```
$ touch hello.asm
```

открываем `mousepad`, заполняем программу

```

1  ; hello.asm
2  SECTION .data ; Секция 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 ; Секция кода
8      GLOBAL _start
9
10     _start: ; Точка входа в программу
11         mov eax,4 ; 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 ; вызов для входа (sys_exit)
18         mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
19         int 80h ; Вызов ядра

```

2. Работа с трансляторами и расширенным синтаксисом.

Превращаю текст программы для вывода "hello world" в объектный код с помощью транслятора NASM. Используем команду `nasm -f elf hello.asm`.

Проверяем правильность выполнения команды.

```
hello.asm hello.o presentation report
```

Обрабатываем компоновщиком LD

```
hello hello.asm hello.o list.lst obj.o presentation report
```

3. Запуск программы

```

$ ./hello
Hello world!

```

4. Выполнение заданий.

Копируем файл `hello_world.asm` в той же директории, переименовывая его как "`Lab04.asm`"

Меняем код так, чтобы вместо `hello world` выводилась наша имя и фамилия

Компилируем файл также как было сделано ранее.

Запускаем программу.

Grobman Alexander

4. Отправляем файлы на гитхаб.

Ссылка на отчёт https://github.com/DaOneme/AEGrobman_study_2023-2024_arhpc/tree/main/Labs/Lab04 (https://github.com/DaOneme/AEGrobman_study_2023-2024_arhpc/tree/main/Labs/Lab04)