МИНОБРНАУКИ РОССИИ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

Кафедра МОЭВМ

ОТЧЕТ

по лабораторной работе №1

по дисциплине «Операционные системы»

Тема: Исследование структур загрузочных модулей

Студент гр. 0381 Павлов Е. А.

Преподаватель Ефремов М. А.

Санкт-Петербург

2022

Цель работы.

Исследование различий в структурах исходных текстов типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.

Задание.

Ассемблерная программа должна читать содержимое предпоследнего байта ROM BIOS, по таблице, сравнивая коды, определять тип PC и выводить строку с названием модели. Если код не совпадает ни с одним значением, то двоичный код переводиться в символьную строку, содержащую запись шестнадцатеричного числа и выводиться на экран в виде соответствующего сообщения.

Затем определяется версия системы. Ассемблерная программа должна по значениям регистров AL и AH формировать текстовую строку в формате xx.yy, где xx – номер основной версии, а yy - номер модификации в десятичной системе счисления, формировать строки с серийным номером OEM и серийным номером пользователя. Полученные строки выводятся на экран.

Результатом выполнения этого шага будет «хороший» .COM модуль, а также необходимо построить «плохой» .EXE, полученный из исходного текста для .COM модуля.

Напишите текст исходного .EXE модуля, который выполняет те же функции и постройте его. Таким образом будет получен хороший .EXE модуль. Сравните исходные тексты для .COM и .EXE модулей. Сравните файлы .COM, «плохого» и «хорошего» .EXE модулей в шестнадцатеричном виде.

Выполнение работы.

Для написания исходного текста .COM модуля был использован шаблон из методических указаний. Были добавлены строки с названиями моделей для последующего вывода на экран.

При запуске программы выполняется переход на метку BEGIN, где происходит считывание байта, расположенного по адресу F000:FFFEh и содержащего информацию о модели компьютера. Затем этот байт последовательно сравнивается с значениями из таблицы в методических указаниях. Если обнаружено совпадение, выводится строка соответствующая данному коду модели, иначе выводится значение в шестнадцатеричном виде.

Для получения информации о версии DOS используется функция 30h прерывания 21h. Полученные значения переводятся в требуемый формат и выводятся на экран.

Результат работы программы:

```
F:\>exe2bin lb1.exe lb1.com

F:\>lb1.com
IBM PC type: AT
MS DOS version: 05.0
OEM:  0
User: 000000H

F:\>
```

Если из этого исходного кода построить .EXE модуль, он будет работать некорректно:

```
F:\>lb1.exe
= Я ъ  н▌╙  Ё  ↑⊖►⊖↑⊖↑⊖⊖⊖⊖ ⊖                    И⊖ш в⊖¶ ↑ Т⊖        ♠

                                                                щ?⊖IBM PC type: PC

                    5 0
               щ?⊖IBM PC type: PC
      0                                                              щ?⊖IBM PC typ
e: PC
                                                          щ?⊖IBM PC type: PC

F:\>
```

Для того, чтобы построить правильный .EXE модуль необходимо разделить программу на сегменты. Для этого в начале исходного текста добавляется описание сегмента стека, а данные и код помещаются в собственные сегменты. Собранный из этого кода .EXE модуль выводит информацию о системе так же, как и .COM модуль.

```
F:\>lb1_exe.exe
IBM PC type: AT
MS DOS version: 05.0
OEM:240
User: 000000H

F:\>_
```

Ответы на вопросы см. в разделе «Вопросы».

Выводы.

Были исследованы различия в структуре исходных текстов для модулей .COM и .EXE, структура загрузочных файлов этих типов и способы загрузки их в основную память.

ВОПРОСЫ

Отличия исходных текстов COM и EXE программ

1. Сколько сегментов должна содержать COM-программа?

Один(сегмент кода), остальные будут сгенерированы автоматически.

2. EXE-программа?

Всего может быть три сегмента, однако допустимо объединение сегментов(например, выделение памяти под стек в сегменте кода)

3. Какие директивы должны обязательно быть в тексте COM-программы?

Смещение(ORG 100h), указание сегментов(ASSUME)

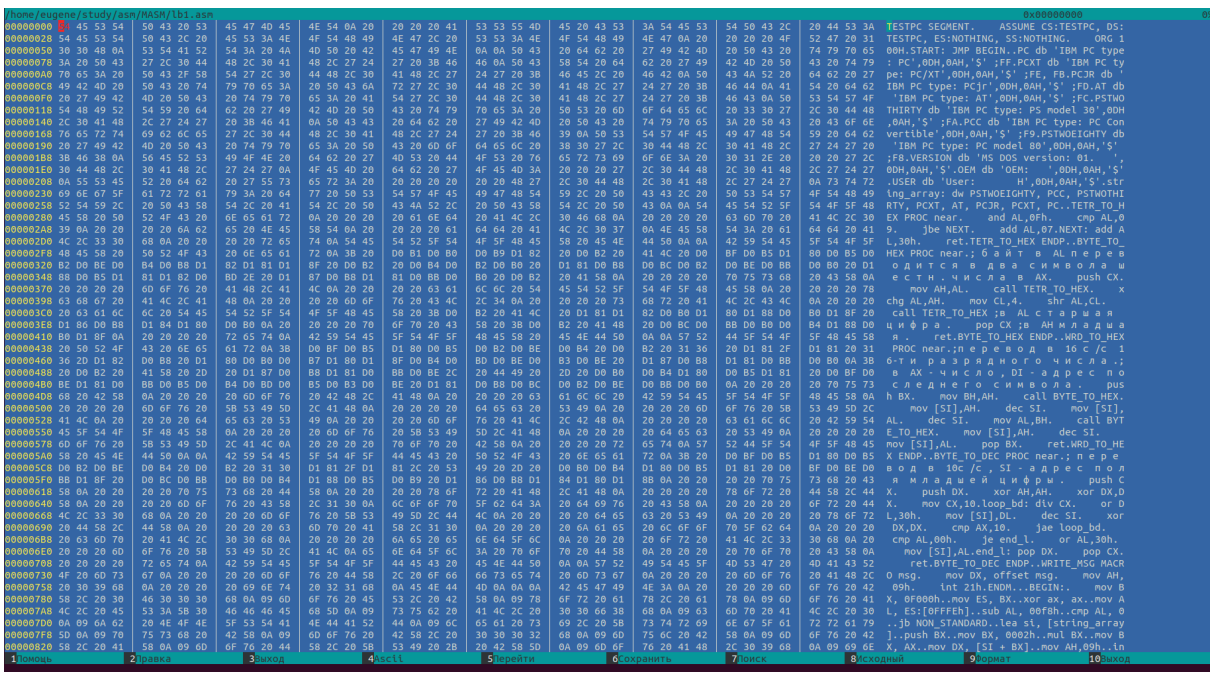
4. Все ли форматы команд можно использовать в COM-программе?

Нет, операции, использующие сегменты запрещены, так как в COM-модуле отсутствует таблица настройки(relocation table).

Отличия форматов файлов COM и EXE модулей

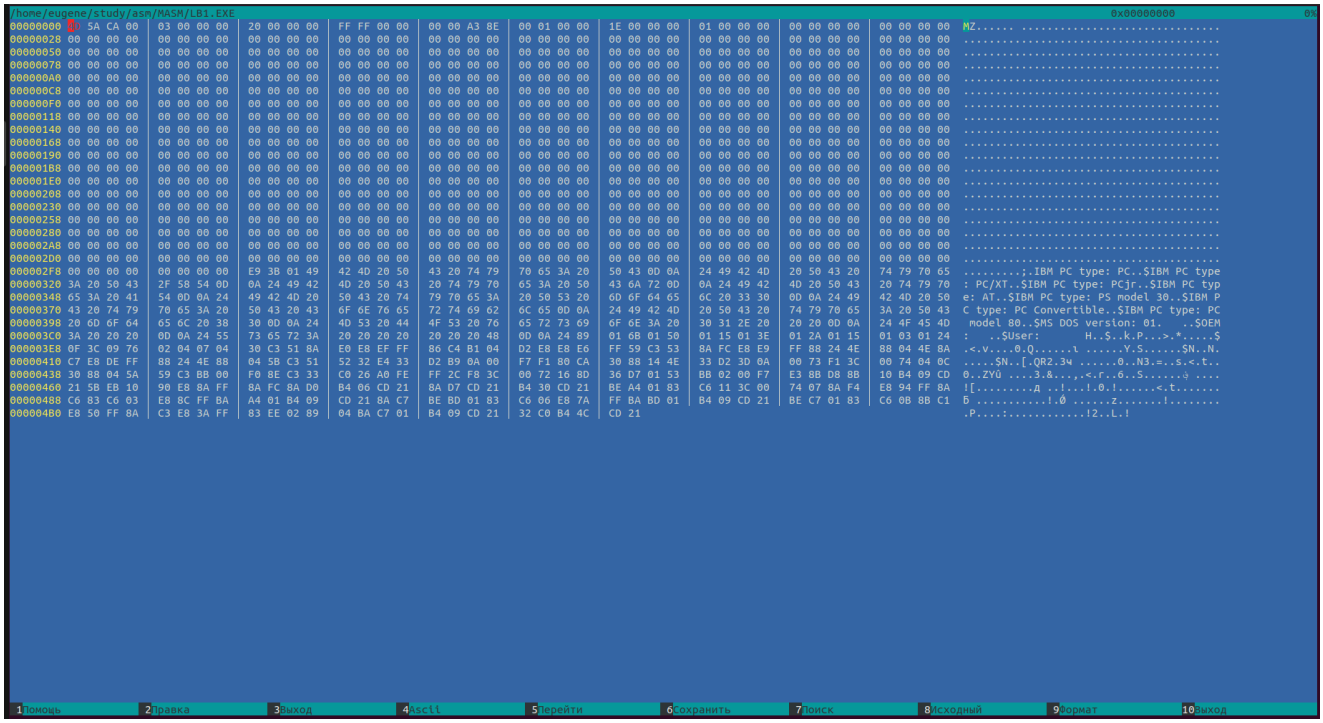
1. Какова структура файла COM? С какого адреса располагается код?

COM файл содержит один сегмент с данными и кодом, размер не превышает 64 Кб. Код начинается с адреса 0h, но при загрузке устанавливается смещение 100h.



2. Какова структура файла «плохого» EXE? С какого адреса располагается код? Что располагается с адреса 0?

Такой файл содержит заголовок, таблицу настройки адресов и один сегмент, в котором находятся данные и код. Код располагается с адреса 300h , потому что 200h – заголовок и таблица настроек, а 100h – смещение ORG 100h. С адреса 0h располагается заголовок EXE файла, который содержит сигнатуру EXE файла, длину образа программы, размер таблицы настройки, сегментный адрес стека, адрес точки входа, а также ряд других параметров, необходимых для загрузки.



3. Какова структура файла «хорошего» EXE? Чем он отличается от файла «плохого» EXE.

«Хороший» содержит заголовок и таблицу настройки адресов, общая длина которых 200h. В отличие от «плохого» EXE файла после таблицы идет три отдельных сегмента: сегмента стека, сегмента данных и сегмент кода.

Загрузка COM модуля в основную память

1. Какой формат загрузки модуля COM? С какого адреса располагается код?

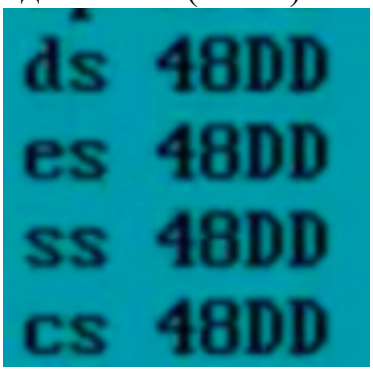
Происходит выделение свободного сегмента памяти, адрес заносится в сегментные регистры. В первые 256 байт этого сегмента записывается PSP, далее происходит подгрузка COM-файла без изменений. В стек записывается адрес возврата, SP указывает на конец сегменты.

2. Что располагается с адреса 0?

Префикс программного сегмента.

3. Какие значения имеют сегментные регистры? На какие области памяти они указывают?

Одинаковое(48DD). Указывают на начало PSP.



4. Как определяется стек? Какую область памяти он занимает? Какие адреса?

Автоматически. Перед PSP. В диапозоне FFFEh-0h

Загрузка «хорошего» EXE модуля в основную память

1. Как загружается «хороший» EXE? Какие значения имеют сегментные регистры?

Построение PSP, определение сегментного адреса, обработка таблицы настройки адресов

2. На что указывают регистры DS и ES?

На PSP.

3. Как определяется стек?

Стек задается парой регистров SS:SP. При запуске программы в SS помещается смещение сегмента стека относительно начального сегмента программы, который содержится в заголовке, плюс адрес начального сегмента. В SP же помещается значение напрямую из заголовка.

4. Как определяется точка входа?

С помощью директивы END