МИНОБРНАУКИ РОССИИ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №1

по дисциплине «**Операционные системы**»

Тема: Исследование структур загрузочных модулей

Студент гр. 6381		Дайнович А.Ю.
Преподаватель		Губкин А.Ф.

Санкт-Петербург

2018

Цель работы:

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

Общие сведения:

Тип IBM PC хранится в байте по адресу 0F000:0FFFE, в предпоследнем байте ROM BIOS. Соответствие кода и типа в таблице:

PC	FF
PC/XT	FE,FB
AT	FC
PS2 модель 30	FA
PS2 модель 50 или 60	FC
PS2 модель 80	F8
PCjr	FD
PC Convertible	F9

Для определения **версии MS DOS** следует воспользоваться функцией 30H прерывания 21H. Входным параметром является номер функции в AH:

```
MOV AH,30h
INT 21h
```

Выходными параметрами являются:

AL – номер основной версии. Если 0, то <2.0;
AH – номер модификации;
BH – серийный номер OEM (Original Equipment Manufacturer);
BL:CX – 24-битовый серийный номер пользователя;

Ход работы:

Шаг 1. Был написан текст исходного .COM модуля, который определяет тип PC и версию системы. Ассемблерная программа должна читать содержимое предпоследнего байта ROM BIOS по таблице, сравнивая коды, определять тип PC и выводить строку с названием модели.

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.

Был отлажен полученный исходный модуль.

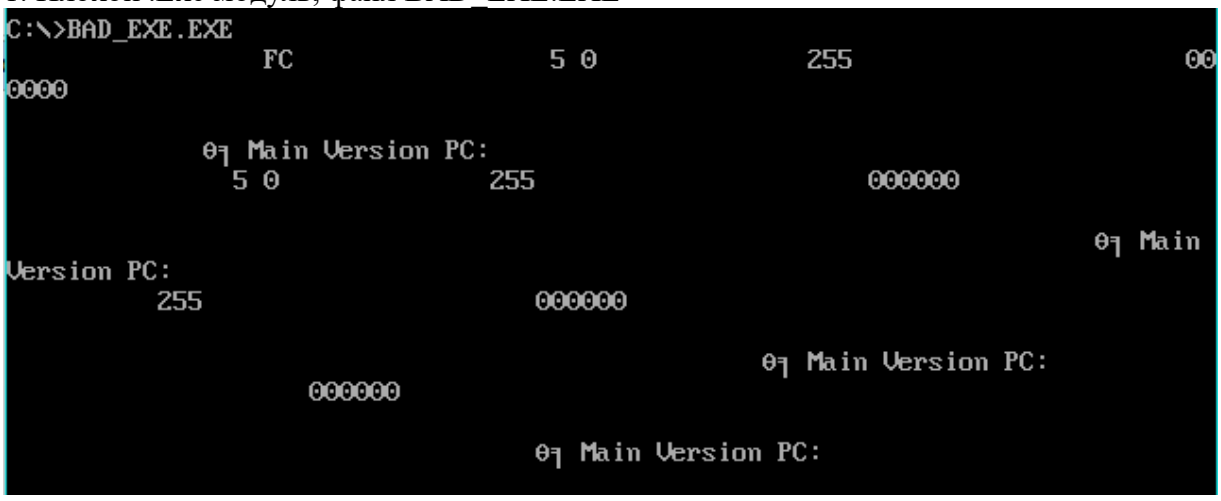
Результатом выполнения этого шага будет «хороший» .COM модуль, а также необходимо построить «плохой» .EXE, полученный из исходного текста для .COM модуля.

Шаг 2. Был написан текст исходного .EXE модуля, который выполняет те же функции, что и модуль в Шаге. Таким образом, будет получен «хороший» .EXE.

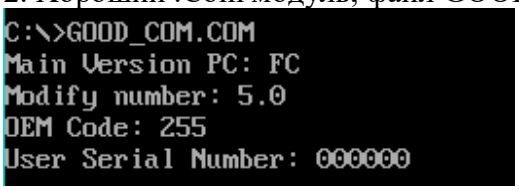
Шаг 3. Были даны ответы на контрольные вопросы.

В результате выполнения лабораторной работы была написана программы, результаты работы которых показаны на рисунках ниже.

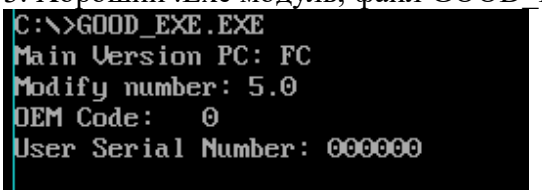
1. Плохой .Exe модуль, файл BAD_EXE.EXE



2. Хороший .Com модуль, файл GOOD_COM.COM



3. Хороший .Exe модуль, файл GOOD_EXE.EXE



Контрольные вопросы:

1. Отличия исходных текстов COM и EXE программ.

а) Сколько сегментов должна содержать COM программа?

.COM - программы содержат только один сегмент.

б) EXE программа?

.EXE-программы отличаются от .COM-программ. В таких программах могут быть сегменты и для кода, и для данных, и для стека.

в) Какие директивы должны обязательно быть в тексте COM программы?

Директива ORG 100h. Нужна для того, чтобы при загрузке модуля в оперативную память в начале COM-программы определялся 256-байтовый (100h) префикс программного сегмента, так что адресация имела смещение в 256 байт от нулевого адреса.

Директива ASSUME. Нужна для того, чтобы задать значения сегментных регистров перед началом работы программы.

Закомментируем директиву ASSUME(В данной программе она указывает, что CS и DS установлены в сегмент TESTPC).

При компиляции возникает ошибка error A2062: Missing or unreachable, так как теперь сегментный регистр не указывает на текущий кодовый сегмент (программа не может найти начало сегмента кода).

г) Все ли форматы команд можно использовать в COM программе?
Нельзя использовать команды, связанные с адресом сегмента, потому что адрес сегмента до загрузки неизвестен. В итоге загрузчик не сможет его определить. Также нельзя использовать оператор FAR - переход на метку возможен только в результате межсегментной передачи управления, а так как в .com-файле только один сегмент, то никаких межсегментных переходов быть не может.

2. Отличия форматов файлов COM и EXE модулей.
а) Какова структура файла COM? С какого адреса располагается код?
.COM-файл состоит из команд, процедур и данных, используемых в программе. Код начинается с нулевого адреса.
б) Какова структура файла «плохого» EXE? С какого адреса располагается код? Что располагается с 0 адреса?
Файл .EXE содержит сегмент PSP, после него идёт основной сегмент памяти. Заголовок располагается с адреса 0h. Код располагается с адреса 300h. С адреса 0 располагается PSP.

в) Какова структура файла «хорошего» EXE? Чем он отличается от «плохого» EXE файла?

EXE-файл состоит из информации для загрузчика, сегмента стека, сегмент данных и сегмент кода. Отличается количеством сегментов (в «плохом» .EXE – 1 сегмент, в хорошем - 3), а также набором разрешённых команд. Код «хорошего» EXE-файла начинается с адреса 200h

3. Загрузка COM модуля в основную память.

а) Какой формат загрузки COM модуля? С какого адреса располагается код?

После загрузки COM-программы в память сегментные регистры указывают на начало PSP. Код располагается с адреса 100h.

б) Что располагается с 0 адреса?

С нулевого адреса располагается заголовок PSP.

в) Какие значения имеют сегментные регистры? На какие области памяти они указывают?

Все сегментные регистры имеют значения 119C. Они указывают на PSP.

г) Как определяется стек? Какую область памяти он занимает? Какие адреса?

Стек создается автоматически, указатель стека в конце сегмента. Из этого следует, что он занимает оставшуюся память и адреса изменяются от больших к меньшим, то есть от FFFEh к 0000h.

4. Загрузка «хорошего» EXE модуля в память.

а) Как загружается «хороший» EXE? Какие значения имеют сегментные регистры?

Сначала создается PSP. Затем определяется длина тела загрузочного модуля, определяется начальный сегмент. Загрузочный модуль считывается в начальный сегмент, таблица настройки считывается в рабочую память, к полю каждого сегмента прибавляется

сегментный адрес начального сегмента, определяются значения сегментных регистров. DS и ES указывают на начало PSP (119C), CS – на начало сегмента команд (11F2h), а SS – на начало сегмента стека (11AC).

б) На что указывают регистры DS и ES?

Изначально регистры DS и ES указывают на начало сегмента PSP.

в) Как определяется стек?

Регистры SS и SP принимают значения, указанные в заголовке, потом к SS прибавляется сегментный адрес начального сегмента.

г) Как определяется точка входа?

Смещение точки входа в программу загружается в указатель команд IP . IP, а именно адрес, с которого начинается выполнение программы, определяется операндом директивы END, который называется точкой входа.