**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. 6383 |  | Липендин А.А. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

**1. Постановка задачи**

* 1. **Цель работы**

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

* 1. **Теоретические 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;

* 1. **Выполнение работы**

В результате выполнения работы были созданы три файла:

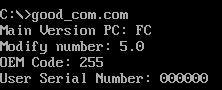
*GOOD\_EXE.exe* - “Хороший” *.exe* модуль

*GOOD\_COM.com -* “Хороший” *.com* модуль

*BAD\_EXE.exe - “Плохой” .exe модуль*

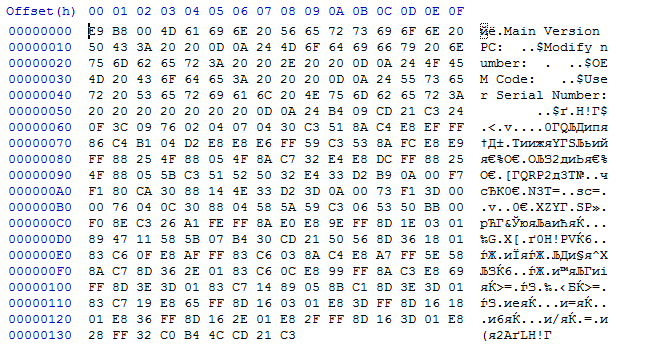
1. **Результаты работы программы**
   1. **Результаты работы с СOM файлом**

* Результат выполнения good\_com.com файла (Рис. 1):



*Рис. 1 Результат выполнения good\_com.com файла*

* Шестнадцатеричный вид загрузочного COM модуля (Рис. 2):



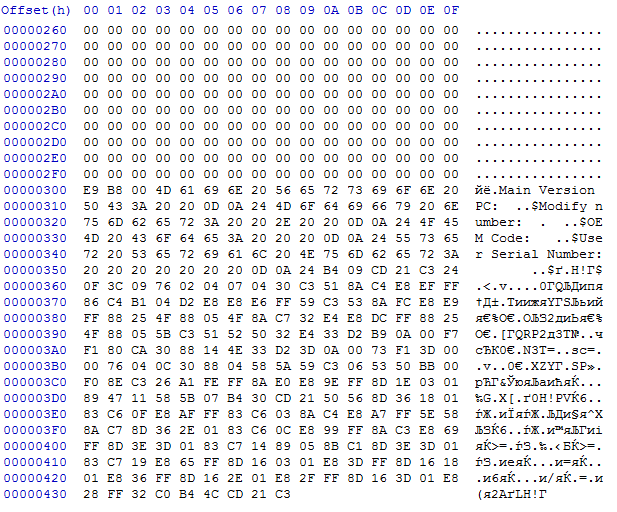
*Рис. 2 Шестнадцатеричный вид загрузочного COM модуля*

* 1. **Результаты работы с “плохим” EXE файлом**
* Результат выполнения bad\_exe.exe файла (Рис. 3):

****

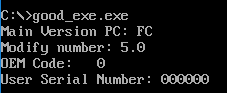
*Рис. 3 Результат выполнения bad\_exe.exe файла*

* Шестнадцатеричный вид “плохого” загрузочного модуля (Рис. 4):

****

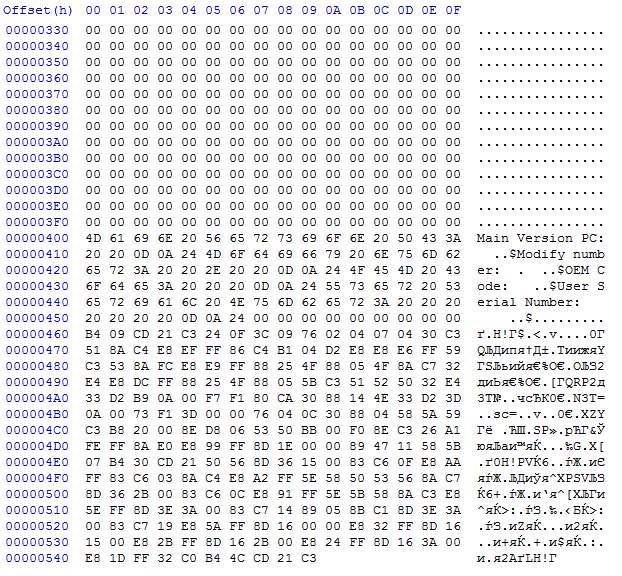
*Рис. 4 Шестнадцатеричный вид “плохого” загрузочного модуля*

* 1. **Результаты работы с «хорошим» EXE файлом**
* Результат работы “хорошего” EXE файла (Рис. 5):



*Рис. 5 Результат работы «хорошего» EXE файла*

* Шестнадцатеричный вид “хорошего” загрузочного модуля (Рис. 6):



*Рис. 6 Шестнадцатеричный вид “хорошего” загрузочного модуля*

1. **Ответы на контрольные вопросы**

**Отличия исходных текстов СОМ и ЕХЕ программ**

1. *Сколько сегментов должна содержать СОМ-программа?*

СОМ-программа содержит один сегмент.

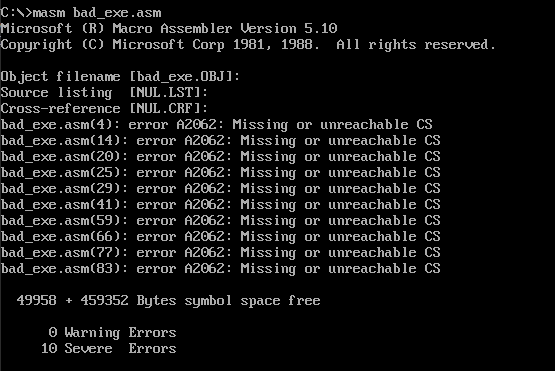
1. *ЕХЕ-программа?*

ЕХЕ-программа может содержать несколько сегментов. В нашем случае их три: сегмент стека, сегмент данных, сегмент кода.

1. *Какие директивы должны быть обязательно в тексте СОМ-программы?*

Все COM-программы должны начинаться с директивы ORG 100h (она устанавливает значение программного счетчика в 100h), потому что при загрузке СОМ-файла в память DOS занимает первые 256 байт (100h) блоком данных PSP, и после этого блока располагает код программы

Также должна присутствовать директива ASSUME, так как ассемблеру необходимо знать о сегменте кода для того, чтобы установить выполняемую программу. Если директива ASSUME будет закомментирована, то при компиляции выявится ошибка о том, что регистр CS не указывает на начало сегмента кода (Рис. 7).



*Рис. 7 Ошибка компиляции при комментировании директивы ASSUME*

1. *Все ли форматы команд можно использовать в СОМ-программе?*

Не все, нельзя применять:

* команды, связанные с адресом сегмента, т.к. адрес сегмента не известен до момента загрузки его в память. Попытка загрузить в регистр адрес сегмента приводит к ошибке.
* Оператор FAR - переход на метку возможен только в результате межсегментной передачи управления, так как в COM-программах есть только один сегмент, то переход между сегментами невозможен.

Файл EXE-формата содержит специальный заголовок, при помощи которого загрузчик выполняет настройку ссылок на сегменты в загруженном модуле.

**Отличия форматов файлов СОМ и ЕХЕ модулей**

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

Файл типа COM состоит из команд, процедур и данных, используемых в программе, код в данном файле начинается с адреса 100h.

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

В файле EXE содержится информация для загрузчика, данные и код (в одном сегменте). Код располагается с адреса 300h. С 0 адреса располагается управляющая информация для загрузчика.

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

“Хороший” 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). Директива ORG не требуется, так как загрузчик автоматически положит программу после PSP.

**Загрузка СОМ модуля в основную память**

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

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

1. *Что располагается с адреса 0?*

С адреса 0 располагается заголовок PSP.

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

Все сегментные регистры имеют значение 119С. Это начало PSP.

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

Адрес верхнего элемента стека хранится в регистре SP (указатель на стек), сегментный адрес стека – в регистре SS. Cтек генерируется автоматически, указатель стека установлен на конец сегмента. Он занимает оставшуюся память и адреса изменяются от больших к меньшим, то есть от FFFEh к 0000h.

**Загрузка «хорошего» ЕХЕ модуля в основную память**

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

В области памяти строится PSP, стандартная часть заголовка считывается в память, 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 (119Ch), CS – на начало сегмента команд (11F7h), а SS – на начало сегмента стека (11ACh).

1. *На что указывают регистры DS и ES?*

На начало PSP.

1. *Как определяется стек?*

Стек определяется при объявлении сегмента стека, в котором указывается, сколько памяти необходимо выделить (в данной работе 256 слов).

1. *Как определяется точка входа?*

Точка входа определяется с помощью директивы end, операндом которой является адрес, с которого начинается выполнение программы.

1. **Вывод**

В ходе работы над программой и поиска ответов на поставленные вопросы были исследованы различия в структурах исходных текстов модулей .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.