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. 9384 _____ Прашутинский К.И.

Преподаватель _____ Ефремов М.А.

Санкт-Петербург

2021

Постановка задачи.

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

Название процедуры	Предназначение процедуры
TETR_TO_HEX	Переводит значение тетрады (4-ех младших битов регистра AL) в цифру 16-ичной СС и представляет ее в виде символа, который далее записывается в регистр AL.
BYTE_TO_HEX	Переводит значение байта (регистра AL) в число 16ичной СС и представляет его в виде двух символов, которые далее записываются в регистры AL и AH.
WRD_TO_HEX	Переводит значение слова (регистра AX) в число 16ичной СС и представляет его в виде четырех символов, которые далее записываются по адресу, на который указывает DI.
BYTE_TO_DEC	Переводит значение байта (регистра AL) в число 10ичной СС и представляет его в виде символов, которые далее записываются по адресу, на который указывает SI.
PRINT	Вызывает функцию вывода строки на экран (функция 09h прерывания 21h).
PC_TYPE	Печатает на экран тип ПК. Если версия ПК не идентифицирована, выводится байт в 16-ичной СС, который содержит информацию о типе ПК.
PC_VERSION	Печатает на экран версию ОС, серийные номера OEM и пользователя.

Последовательность действий программы.

1)Вызывается процедура PC_TYPE, которая выводит на экран тип ПК. Информация о типе ПК получается из предпоследнего байта ROM BIOS по адресу 0F000:0FFFEh. В зависимости от значения байта определяется тип:

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

В случае, если значение не совпадает со значениями, приведенными выше, то в качестве типа ПК выводится значения предпоследнего байта ROM BIOS.

2)Вызывается процедура PC_VERSION, которая выводит на экран версию ОС, серийные номера OEM и пользователя. Информация получается при помощи вызова функции 30h прерывания 21h.

3)Завершение работы программы.

Ход работы.

1. Для начала был написан текст исходного .COM модуля lab1_com.asm. Далее при помощи транслятора TASM.EXE и компоновщика TLINK.EXE был скомпилирован «хороший» .COM и «плохой» .EXE модуль lab1_com.exe.
2. Отладил «плохой» .EXE модуль и по нему построил «хороший» .EXE модуль
3. Сравнил исходные тексты lab1_com.asm и lab1_exe.asm.
4. После при помощи программы FAR были открыты файлы загрузочных модулей lb1_COM.com, lb1_EXE.exe и lb1_COM.exe в шестнадцатеричном виде, далее было выполнено сравнение.
5. Далее был исследован загрузочный модуль .COM при помощи отладчика TD.EXE.
6. Далее был исследован «хороший» загрузочный модуль .EXE при помощи отладчика TD.EXE.

Результаты исследования проблем.

План загрузки модуля .COM в основную память.

1. Определяется сегментный адрес свободного участка основной памяти для загрузки программы.
2. Для программы формируется блок PSP и загружается в начало.
3. После блока PSP по адресу PSP:0100h загружается COM модуль.

После загрузки COM-программы в основную память CS, DS, ES и SS указывают на PSP, SP указывает на конец сегмента PSP, слово 00h помещено на стек, IP имеет значение 100h.

Контрольные вопросы.

Отличия исходных текстов COM и EXE программ.

1. Сколько сегментов должна содержать COM-программа?

   Только один сегмент
2. Сколько сегментов должна содержать EXE-программа?

   Один или более сегментов.
3. Какие директивы должны обязательно быть в тексте COM-программы?

Директива ORG. В начальной части COM-программы размещается специальный блок PSP (префикс программного сегмента), в начале которого размещена команда вызова обработчика прерывания для завершения программы и возврата в DOS. Так как после загрузки все сегментные регистры, включая CS, указывают на начало PSP, а IP = 0, то программа не может исполняться, начиная с этого адреса. Поэтому требуется директива ORG 100h, которая установит CS:IP на конец PSP (размер PSP равен 256 байт, значит сдвиг нужно делать на 256 = 100h байт).

Директива ASSUME. Указывает ассемблеру, с каким сегментом или группой сегментов связан тот или иной сегментный регистр. Она не изменяет значений сегментных регистров, а только позволяет ассемблеру проверять допустимость ссылок и самостоятельно вставлять при необходимости

префиксы переопределения сегментов. Без этой директивы программа не скомпилируется, так как ассемблер не будет понимать, относительно чего вычислять смещения меток.

Директива END. Этой директивой завершается любая программа на ассемблере.

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

Нет. Так как в COM программах в DOS нет таблицы настройки, которая содержит описание адресов, зависящих от размещения загрузочного модуля в ОП, потому что подобные адреса в нем запрещены. Поэтому нельзя использовать команды, связанные с адресом сегмента, потому что адрес сегмента до загрузки неизвестен.

Отличия форматов файлов COM и EXE модулей.

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

В COM-файле код, данные и стек располагаются в одном сегменте. Код (как и данные) начинается с адреса 0h. Файл в 16-ичном представлении расположен снизу.

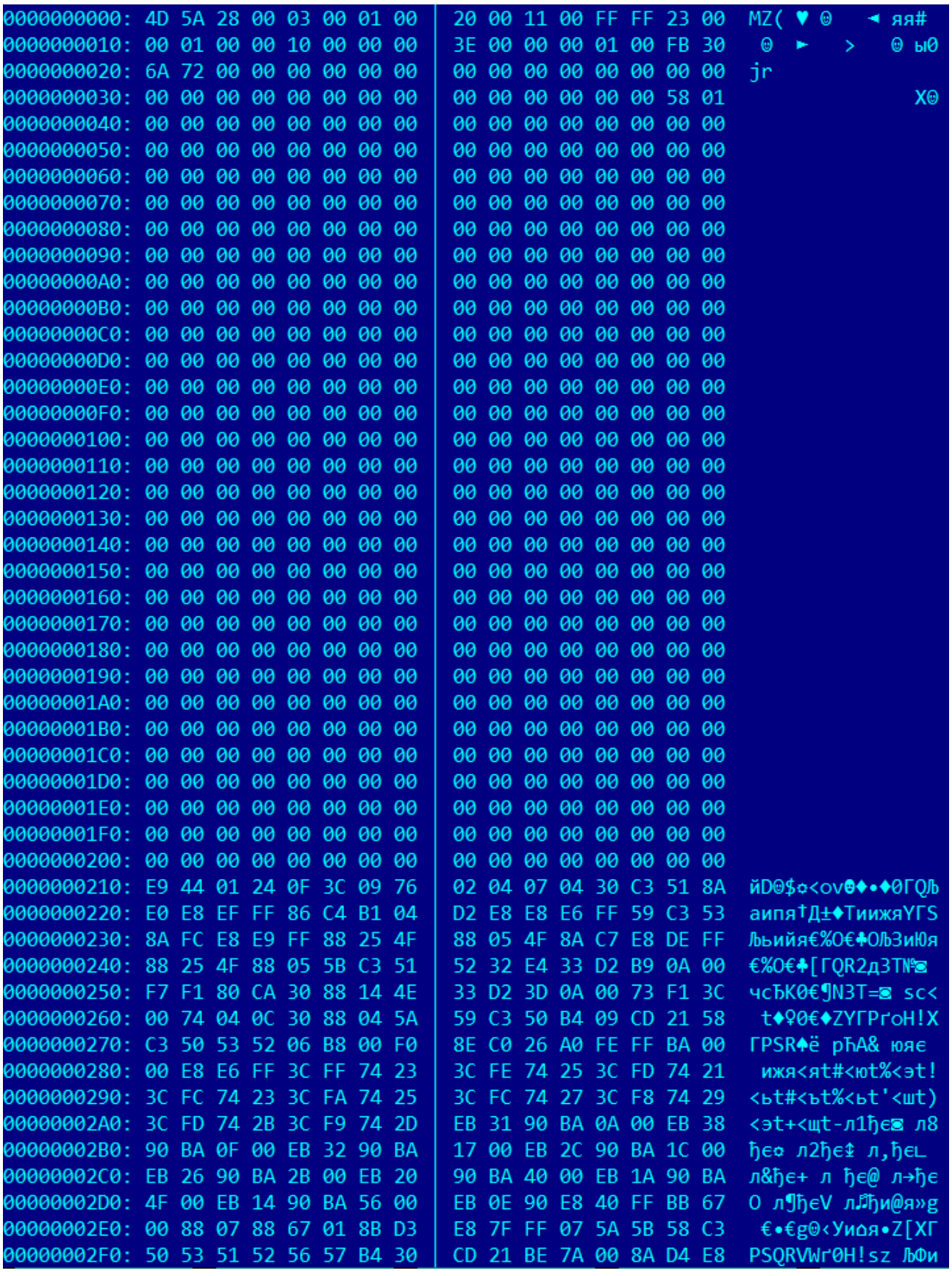
2. Какова структура файла «плохого» EXE? С какого адреса располагается код? Что располагается с адреса 0?

   В плохом EXE-файле код, данные и стек располагаются в одном сегменте. Код (как и данные) начинается с адреса 300h. С адреса 0h располагается управляющая информация для загрузчика, которая содержит заголовок и таблицу настройки адресов. Файл в 16-ичном представлении расположен снизу.

```
0000000000: 4D 5A 0B 01 03 00 00 00  20 00 00 00 FF FF 00 00  MZ♂☺♥       ÿÿ
0000000010: 00 00 00 00 00 01 00 00  3E 00 00 00 01 00 FB 30        ☺  >    ☺ û0
0000000020: 6A 72 00 00 00 00 00 00  00 00 00 00 00 00 00 00  jr
0000000030: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000040: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000050: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000060: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000070: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000080: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000090: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000100: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000110: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000120: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000130: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000140: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000150: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000160: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000170: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000180: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000190: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000200: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000210: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000220: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000230: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000240: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000250: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000260: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000270: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000280: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000290: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000002F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
```

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

   В хорошем EXE-файле код, данные и стек находятся в различных сегментах, а в плохом – в одном и том же сегменте. По мимо этого, изменен порядок расположения сегментов в памяти, и стек имеет другой размер – 256 байт (100h). Также разница между «хорошим» EXE и «плохим» EXE в том, что «хороший» не имеет директивы 100h и поэтому не выделяет память под PSP, а значит смещение у хорошего 210h, а у плохого 300h. Файл в 16-ичном представлении расположен снизу.



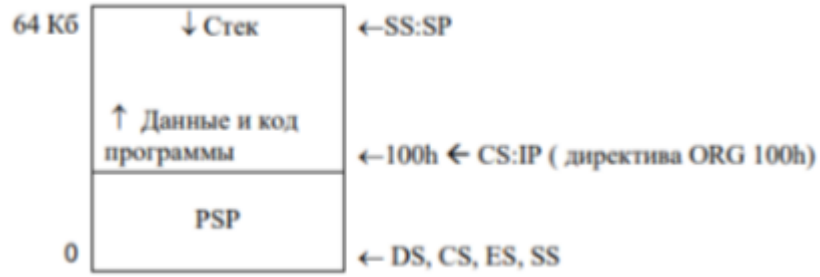
```
0000000000: 4D 5A 28 00 03 00 01 00  20 00 11 00 FF FF 23 00  MZ( ♥ ☺   ◄ яя#
0000000010: 00 01 00 00 10 00 00 00  3E 00 00 00 01 00 FB 30   ☺ ►   >   ☺ ы0
0000000020: 6A 72 00 00 00 00 00 00  00 00 00 00 00 00 00 00  jr
0000000030: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 58 01                X☺
0000000040: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000050: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000060: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000070: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000080: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000090: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000000F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000100: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000110: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000120: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000130: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000140: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000150: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000160: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000170: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000180: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000190: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000001F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000200: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0000000210: E9 44 01 24 0F 3C 09 76  02 04 07 04 30 C3 51 8A  йD☺$☼<ov☻♦•♦0ГQЉ
0000000220: E0 E8 EF FF 86 C4 B1 04  D2 E8 E8 E6 FF 59 C3 53  аипя†Д±♦ТиижяYГS
0000000230: 8A FC E8 E9 FF 88 25 4F  88 05 4F 8A C7 E8 DE FF  Љьийя€%O€♣OЉЗиЮя
0000000240: 88 25 4F 88 05 5B C3 51  52 32 E4 33 D2 B9 0A 00  €%O€♣[ГQR2д3ТЊ◙
0000000250: F7 F1 80 CA 30 88 14 4E  33 D2 3D 0A 00 73 F1 3C  чсЂK0€¶N3T=◙ sc<
0000000260: 00 74 04 0C 30 88 04 5A  59 C3 50 B4 09 CD 21 58   t♦♀0€♦ZYГPґoH!X
0000000270: C3 50 53 52 06 B8 00 F0  8E C0 26 A0 FE FF BA 00  ГPSR♠ё рЋA& юяє
0000000280: 00 E8 E6 FF 3C FF 74 23  3C FE 74 25 3C FD 74 21   ижя<ят#<ют%<эt!
0000000290: 3C FC 74 23 3C FA 74 25  3C FC 74 27 3C F8 74 29  <ьt#<ъt%<ьt'<шt)
00000002A0: 3C FD 74 2B 3C F9 74 2D  EB 31 90 BA 0A 00 EB 38  <эt+<щt-л1ђє◙ л8
00000002B0: 90 BA 0F 00 EB 32 90 BA  17 00 EB 2C 90 BA 1C 00  ђє☼ л2ђє↕ л,ђєL
00000002C0: EB 26 90 BA 2B 00 EB 20  90 BA 40 00 EB 1A 90 BA  л&ђє+ л ђє@ л→ђє
00000002D0: 4F 00 EB 14 90 BA 56 00  EB 0E 90 E8 40 FF BB 67  O л¶ђєV л♫ђи@я»g
00000002E0: 00 88 07 88 67 01 8B D3  E8 7F FF 07 5A 5B 58 C3   €•€g☺‹Уиоя•Z[XГ
00000002F0: 50 53 51 52 56 57 B4 30  CD 21 BE 7A 00 8A D4 E8  PSQRVWґ0H!sz ЉФи
```

```
0000000300: 45 FF 8A C2 83 C6 03 E8  3D FF BA 6C 00 E8 5A FF  EяЉВѓЖ♥и=яєl иZя
0000000310: BA 79 00 E8 54 FF 8A C7  E8 03 FF BF 95 00 88 05  єy иTяЉЗи♥яї• €♣
0000000320: 88 65 01 BA 81 00 E8 41  FF BA 95 00 E8 3B FF 8A  €e☺єЃ иАяє• и;яЉ
0000000330: C3 E8 EA FE BF AF 00 88  05 88 65 01 8B C1 83 C7  Гикюїї €♣€e☺‹Бѓ3
0000000340: 05 E8 EB FE BA 9A 00 E8  20 FF BA AF 00 E8 1A FF  ♣илюєь и яєї и→я
0000000350: 5F 5E 5A 59 5B 58 C3 B8  17 00 8E D8 E8 12 FF E8  _^ZY[XГё↕ ЋШи↕яи
0000000360: 8E FF 32 C0 B4 4C CD 21  00 00 00 00 00 00 00 00  Ћя2АґLH!
0000000370: 50 43 20 74 79 70 65 3A  20 24 50 43 0D 0A 24 50  PC type: $PC♪◙$P
0000000380: 43 2F 58 54 0D 0A 24 41  54 0D 0A 24 50 53 32 20  C/XT♪◙$AT♪◙$PS2
0000000390: 6D 6F 64 65 6C 20 33 30  0D 0A 24 50 53 32 20 6D  model 30♪◙$PS2 m
00000003A0: 6F 64 65 6C 20 35 30 20  6F 72 20 36 30 0D 0A 24  odel 50 or 60♪◙$
00000003B0: 50 53 32 20 6D 6F 64 65  6C 20 38 30 0D 0A 24 50  PS2 model 80♪◙$P
00000003C0: 43 6A 72 0D 0A 24 50 43  20 43 6F 6E 76 65 72 74  Cjr♪◙$PC Convert
00000003D0: 69 62 6C 65 0D 0A 24 20  20 0D 0A 24 4F 53 20 76  ible♪◙$  ♪◙$OS v
00000003E0: 65 72 73 69 6F 6E 3A 20  24 20 20 2E 20 20 0D 0A  ersion: $  .  ♪◙
00000003F0: 24 4F 45 4D 20 73 65 72  69 61 6C 20 6E 75 6D 62  $OEM serial numb
0000000400: 65 72 3A 20 24 20 20 0D  0A 24 55 73 65 72 20 73  er: $  ♪◙$User s
0000000410: 65 72 69 61 6C 20 6E 75  6D 62 65 72 3A 20 24 20  erial number: $
0000000420: 20 20 20 20 20 0D 0A 24                           ♪◙$
```

Загрузка COM модуля в основную память.

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

   COM-модуль имеет следующую структуру:



   Код располагается с адреса 100h, поскольку с адреса 0h располагается блок PSP размером 256 байт.

```
AX 0000   SI 0000   CS 119C   IP 0100
BX 0000   DI 0000   DS 119C
CX 020B   BP 0000   ES 119C   HS 119C
DX 0000   SP FFF5   SS 119C   FS 119C
```

2. Что располагается с адреса 0?

   Специальный блок PSP (префикс программного сегмента).

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

   Регистры DS, ES, CS, SS указывают на начало блока PSP.

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

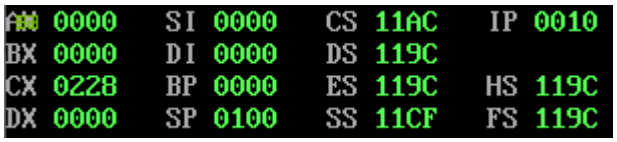
Регистр SS указывает на начало блока PSP (0h), а SP указывает на конец модуля (FFFFh). То есть стек расположен между адресами SS:0000h и SS:FFFFh и заполняется с конца модуля в строну уменьшения адресов.

Загрузка «хорошего» EXE модуля в основную память.

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

   Регистры DS и ES указывают на начало блока PSP, регистр CS указывает на начало сегмента кода, а регистр SS – на начало сегмента стека.

   Структура выглядит следующим образом:



2. На что указывают регистры DS и ES?

   Регистры DS и ES указывают на начало блока PSP.

3. Как определяется стек?

   Регистр SS указывает на начало сегмента стека, а SS:SP – на конец сегмента стека.

4. Как определяется точка входа?

   Точка входа определяется параметром после директивы END, в качестве которого нужно передать метку, с которой программа начнет выполнение команд.

Выводы.

Были изучены различия в структурах исходных текстов модулей типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.