

**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. 7383

\_\_\_\_\_

Бергалиев М.

Преподаватель

\_\_\_\_\_

Ефремов М.А.

Санкт-Петербург

2019

**Цель работы:** исследование различий в структурах исходных текстов модулей типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.

### **Ход работы.**

Использованные функции:

TETR\_TO\_HEX – переводит младшую тетраду битов из регистра AL в шестнадцатеричную систему.

BYTE\_TO\_HEX – переводит число из регистра AL в шестнадцатеричную систему.

WRD\_TO\_HEX – переводит число из регистра AX в шестнадцатеричную систему.

BYTE\_TO\_DEC – переводит число из регистра AL в десятичную систему.

PC\_TYPE – по байту по адресу 0F000:0FFFFh определяет тип IBM PC и записывает в регистр DX смещение строки с необходимым типом.

OS\_VERSION – по значению регистра AX записывает в строку NUM версию MS DOS и помещает в регистр DX смещение строки NUM.

OEM – по значению регистра BH записывает в строку NUM серийный номер OEM и помещает в регистр DX смещение строки NUM.

USER\_NUM – по значениям регистров BL и CX записывает в строку NUM серийный номер пользователя и помещает в регистр DX смещение строки NUM.

Использованные структуры данных:

NUM – строка, в которую записываются преобразованные числа.

Строки, содержащие тип IBM PC: PC, PC\_XT, AT, PS2\_30, PS2\_80, PCjr, PC\_Conv.

PC\_TYPE\_STR – строка, заглавливающая тип IBM PC.

OS\_VERSION\_STR – строка, заглавливающая версию MS DOS.

LESS\_2 – строка, сообщающая, что версия MS DOS меньше 2.0.

OEM\_STR – строка, заглавляющая серийный номер OEM.

USER\_NUM\_STR – строка, заглавляющая серийный номер пользователя.

Программа получает необходимые данные о системе из ROM BIOS и с помощью функции 30h прерывания 21h. Выводит на экран тип IBM PC, версию ОС, серийный номер OEM и серийный номер пользователя.

Результаты работы программы показаны на рис. 1-3.

```
C:\>COM.COM
Тип IBM PC: AT
Версия ОС: 5.0
OEM: 255
Серийный номер пользователя: 0000000
```

Рисунок 1 – Результат выполнения программы COM.COM

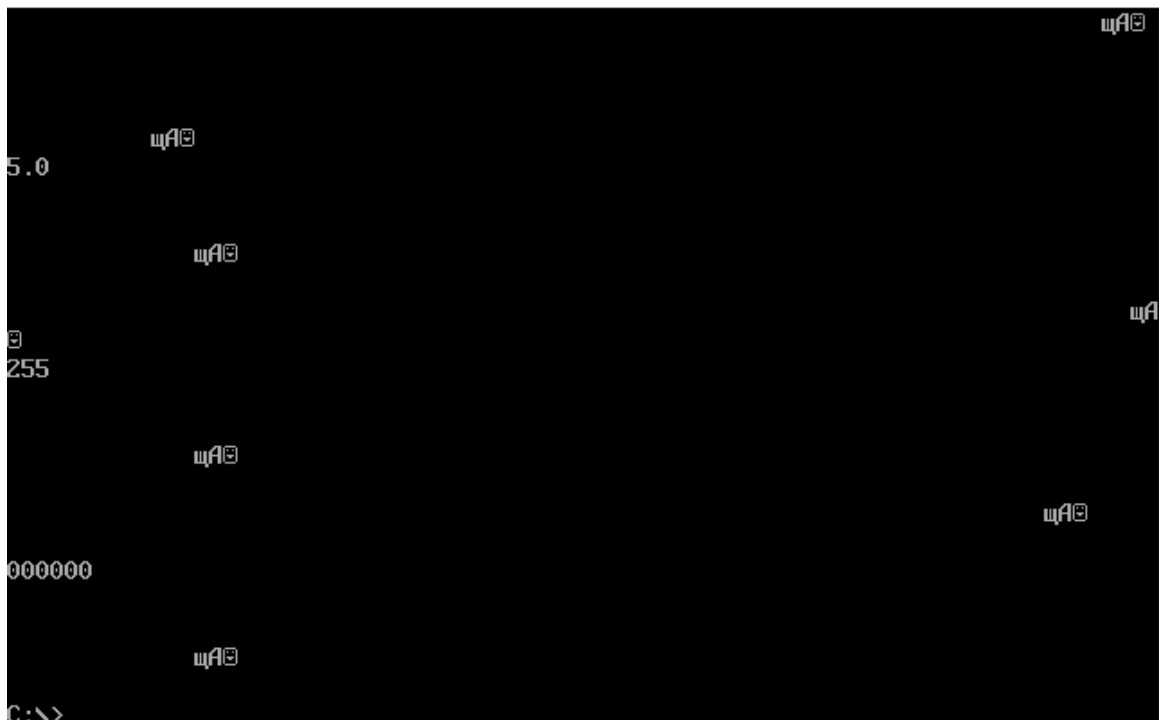


Рисунок 2 — Результат выполнения программы COM.EXE

```
C:\>EXE.EXE
Тип IBM PC: AT
Версия ОС: 5.0
OEM: 255
Серийный номер пользователя: 0000000
```

Рисунок 3 — Результат выполнения программы EXE.EXE

## **Ответы на вопросы:**

### **Отличия исходных текстов COM и EXE программ**

1. Сколько сегментов должна содержать COM-программа?

Ровно один сегмент.

2. EXE-программа?

Один или более.

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

ORG – предписывает место расположения первой команды программы, поскольку в начале программы лежит PSP. ASSUME – ставит сегментным регистрам в соответствие требуемые сегменты.

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

Нельзя использовать команды с дальней адресацией, поскольку в COM-программе отсутствует таблица настроек, которая указывает, какие абсолютные адреса при загрузке должны быть изменены, так как до загрузки неизвестно, куда будет загружена программа.

### **Отличия форматов файлов COM и EXE модулей**

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

Структура файла показана на рис. 4. Код и данные располагаются с нулевого адреса.

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

Структура файла приведена на рис. 5, 6. Код располагается с адреса 300h. С адреса 0 располагается заголовок, таблица настроек, а также зарезервированные директивой ORG 100h байт.

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

Структура файла показана на рис. 7, 8. Как и в «плохом» EXE, с адреса 0 располагается заголовок с таблицей настроек. С адреса 200h идет сегмент стека, когда как в «плохом» EXE здесь располагаются зарезервированные ORG 100h байт и нет сегмента стека. В EXE программах нет необходимости

использовать директиву ORG, поскольку загрузчик ставит программу после PSP. С адреса 220h располагается сегмент данных, с адреса 2C0h — сегмент кода.

0000000000: E9 8E 01 20 20 20 20 20	0D 0A 24 50 43 0D 0A 24	щ00 J\$PCJ\$
0000000010: 50 43 2F 58 54 0D 0A 24	41 54 0D 0A 24 50 53 32	PC/XTJ\$ATJ\$PS2
0000000020: 20 AC AE A4 A5 AB EC 20	33 30 0D 0A 24 50 53 32	модель 30J\$PS2
0000000030: 20 AC AE A4 A5 AB EC 20	38 30 0D 0A 24 50 53 6A	модель 80J\$PSj
0000000040: 72 0D 0A 24 50 53 20 43	6F 6E 76 65 72 74 69 62	rJ\$PS Convertib
0000000050: 6C 65 0D 0A 24 92 A8 AF	20 49 42 4D 20 50 43 3A	leJ\$Тип IBM PC:
0000000060: 20 24 82 A5 E0 E1 A8 EF	20 8E 91 3A 20 24 3C 32	\$Версия ОС: \$<2
0000000070: 0D 0A 24 4F 45 4D 3A EF	24 91 A5 E0 A8 A9 AD EB	J\$OEM: \$Серийны
0000000080: A9 20 AD AE AC A5 E0 20	AF AE AB EC A7 AE A2 A0	й номер пользова
0000000090: E2 A5 AB EF 3A 20 24 24	0F 3C 09 76 02 04 07 04	теля: \$\$<ov0♦♦
00000000A0: 30 C3 51 8A E0 E8 EF FF	86 C4 B1 04 D2 E8 E8 E6	0 QКршя Ж—♦тшщ
00000000B0: FF 59 C3 53 8A FC E8 E9	FF 88 25 4F 88 05 4F 8A	Y SKNщц И%OI+OK
00000000C0: C7 E8 DE FF 88 25 4F 88	05 5B C3 51 52 32 E4 33	ш  И%OI+   QR2ф3
00000000D0: D2 B9 0A 00 F7 F1 80 CA	30 88 14 4E 33 D2 3D 0A	т    üäA±0иJN3т=
00000000E0: 00 73 F1 3C 00 74 04 0C	30 88 04 5A 59 C3 50 06	së< т♦0и♦ZY P+
00000000F0: B8 00 F0 8E C0 26 A0 FE	FF 3C FF 74 1C 3C FE 74	т   EO L&a  < тL<
0000000100: 1E 3C FD 74 20 3C FC 74	22 3C FB 74 12 3C FA 74	▲<тt <N"t" <vt↓<·t
0000000110: 20 3C F9 74 28 3C F8 74	1E BA 0B 01 EB 22 90 BA	<·t(<°t▲ д0ы"Р
0000000120: 10 01 EB 1C 90 BA 3D 01	EB 16 90 BA 18 01 EB 10	►0ыLP  =0ыP  ↑0ы►
0000000130: 90 BA 1D 01 EB 0A 90 BA	2D 01 EB 04 90 BA 44 01	Р  ←0ыP  -0ы♦P  D0
0000000140: 07 58 C3 3C 00 74 17 BE	03 01 E8 7E FF B2 2E 83	·X < т±=♥0ш~  .Г
0000000150: C6 02 88 14 46 8A C4 E8	71 FF BA 03 01 C3 BA 6E	0иJFK-шq   ♥0 тн
0000000160: 01 C3 8A C7 BE 03 01 83	C6 02 E8 5E FF BA 03 01	0 K  д♥0Г 0ш^   ♥0
0000000170: C3 8A C3 BF 03 01 E8 29	FF 88 25 88 45 01 83 C7	K ♥0ш) И%ИЕ0Г
0000000180: 05 8B C1 E8 2D FF BA 03	01 C3 50 B4 09 CD 21 58	♦л-ш-   ♥0 P о= X
0000000190: C3 BA 55 01 E8 F3 FF E8	54 FF E8 ED FF B4 30 CD	U0шє шТ шэ  0=
00000001A0: 21 BA 62 01 E8 E3 FF E8	99 FF E8 DD FF BA 73 01	! b0шу шщ ш    s0
00000001B0: E8 D7 FF E8 AC FF E8 D1	FF BA 79 01 E8 CB FF E8	ш т шм шт   у0шт ш
00000001C0: AF FF E8 C5 FF 32 C0 B4	4C CD 21	п шт 2 L=!

Рисунок 4 — Структура COM файла

0000000000: 4D 5A CF 00 03 00 00 00	20 00 00 00 FF FF 00 00	MZ± ♥
0000000010: 00 00 45 36 00 01 00 00	1E 00 00 00 01 00 00 00	E6 0 ▲ 0
0000000020: 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
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	

Рисунок 5 — Структура «плохого» EXE файла, адреса 0h-240h

000000250:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000260:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000270:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000280:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000290:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002A0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002B0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002C0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002D0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002E0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000002F0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000300:	E9 90 01 20 20 20 20 20	0D 0A 24 50 43 0D 0A 24	щР0        )PC)PC\$
000000310:	50 43 2F 58 54 0D 0A 24	41 54 0D 0A 24 50 53 32	PC/XT)AT)PS2
000000320:	20 AC AE A4 A5 AB EC 20	33 30 0D 0A 24 50 53 32	модель 30)PS2
000000330:	20 AC AE A4 A5 AB EC 20	38 30 0D 0A 24 50 53 6A	модель 80)PSj
000000340:	72 0D 0A 24 50 53 20 43	6F 6E 76 65 72 74 69 62	r)PS Convertib
000000350:	6C 65 0D 0A 24 92 A8 AF	20 49 42 4D 20 50 43 3A	le)Тип IBM PC:
000000360:	20 24 82 A5 E0 E1 A8 EF	20 8E 91 3A 20 24 3C 32	\$Версия ОС: \$<2
000000370:	2E 30 0D 0A 24 4F 45 4D	3A 20 24 91 A5 E0 A8 A9	.0)OEM: \$Серий
000000380:	AD EB A9 20 AD AE AC A5	E0 20 AF AE AB EC A7 AE	ный номер пользо
000000390:	A2 A0 E2 A5 AB EF 3A 20	24 24 0F 3C 09 76 02 04	вателя: \$<ov0
0000003A0:	07 04 30 C3 51 8A E0 E8	EF FF 86 C4 B1 04 D2 E8	♦0 QКршя Ж—Тш
0000003B0:	E8 E6 FF 59 C3 53 8A FC	E8 E9 FF 88 25 4F 88 05	щц Y SKNщц И%OI+
0000003C0:	4F 8A C7 E8 DE FF 88 25	4F 88 05 5B C3 51 52 32	OK  ш   И%OI+   QR2
0000003D0:	E4 33 D2 B9 0A 00 F7 F1	80 CA 30 88 14 4E 33 D2	фЗТ ) ŷ&A±0ИГNЗТ
0000003E0:	3D 0A 00 73 F1 3C 00 74	04 0C 30 88 04 5A 59 C3	=) s&lt; t+?0И+ZY
0000003F0:	50 06 B8 00 F0 8E 0C 26	A0 FE FF 3C FF 74 1C 3C	P+q E0L&a■ < tL<
000000400:	FE 74 1E 3C FD 74 20 3C	FC 74 22 3C FB 74 12 3C	■tΔ<Rt <Nt" <Vt&lt;
000000410:	FA 74 20 3C F9 74 28 3C	F8 74 1E BA 0B 01 EB 22	·t <·t(<°tΔ  δ0ы"
000000420:	90 BA 10 01 EB 1C 90 BA	3D 01 EB 16 90 BA 18 01	P  >0ыLP  =0ы=P  ↑0
000000430:	EB 10 90 BA 1D 01 EB 0A	90 BA 2D 01 EB 04 90 BA	ы>P  <0ы0P  -0ы+P
000000440:	44 01 07 58 C3 3C 00 74	17 BE 03 01 E8 7E FF B2	D0·X < t&±=♥0ш~
000000450:	2E 83 C6 02 88 14 46 8A	C4 E8 71 FF BA 03 01 C3	.Г 0ИГFK-шq   ♥0
000000460:	BA 6E 01 C3 8A C7 BE 03	01 83 C6 02 E8 5E FF BA	n0 K &±♥0Г 0ш^
000000470:	03 01 C3 8A C3 BF 03 01	E8 29 FF 88 25 88 45 01	♥0 K ♥0ш) И%ИE0
000000480:	83 C7 05 8B C1 E8 2D FF	BA 03 01 C3 50 B4 09 CD	Г  +л-ш-   ♥0 P <0=
000000490:	21 58 C3 8C D8 BA 55 01	E8 F1 FF E8 52 FF E8 EB	!X М±  U0ш& шР шы

Рисунок 6 — Структура «плохого» EXE файла, адреса 250h-490h

000000000:	4D 5A 2D 00 03 00 16 00	20 00 00 00 FF FF 00 00	MZ- ▼ -
000000010:	20 00 18 E0 0C 01 0C 00	1E 00 00 00 01 00 11 00	↑p&0& ▲ 0 ◀
000000020:	0C 00 1C 00 0C 00 26 00	0C 00 33 00 0C 00 BE 00	♀ L ♀ & ♀ 3 ♀ ♀
000000030:	0C 00 CD 00 0C 00 E2 00	0C 00 F0 00 0C 00 FF 00	♀ = ♀ т ♀ E ♀
000000040:	0C 00 13 01 0C 00 20 01	0C 00 25 01 0C 00 2A 01	♀ !!0♀ 0♀ %0♀ *0
000000050:	0C 00 36 01 0C 00 3B 01	0C 00 40 01 0C 00 48 01	♀ 60♀ ;0♀ @0♀ H0
000000060:	0C 00 4D 01 0C 00 52 01	0C 00 5A 01 0C 00 5F 01	♀ M0♀ R0♀ Z0♀ _0
000000070:	0C 00 64 01 0C 00 00 00	00 00 00 00 00 00 00 00	♀ d0♀
000000080:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000090:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000A0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000B0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000C0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000D0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000E0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000F0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000100:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000110:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000120:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000130:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000140:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000150:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000160:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000170:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000180:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000190:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001A0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001B0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001C0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001D0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001E0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000001F0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000200:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000210:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
000000220:	20 20 20 20 20 0D 0A 24	50 43 0D 0A 24 50 43 2F	)PC)PC/
000000230:	58 54 0D 0A 24 41 54 0D	0A 24 50 53 32 20 AC AE	XT)AT)PS2 мо
000000240:	A4 A5 AB EC 20 33 30 0D	0A 24 50 53 32 20 AC AE	дель 30)PS2 мо

Рисунок 7 — Структура «хорошего» EXE файла, адреса 0h-240h



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:	20 20 20 20 20 0D 0A 24	50 43 0D 0A 24 50 43 2F	PC/PC/
0000000230:	58 54 0D 0A 24 41 54 0D	0A 24 50 53 32 20 AC AE	XTJ\$AT\$PS2 мо
0000000240:	A4 A5 AB EC 20 33 30 0D	0A 24 50 53 32 20 AC AE	дель 30\$PS2 мо
0000000250:	A4 A5 AB EC 20 38 30 0D	0A 24 50 53 6A 72 0D 0A	дель 80\$PSjr
0000000260:	24 50 53 20 43 6F 6E 76	65 72 74 69 62 6C 65 0D	\$PS Convertible
0000000270:	0A 24 92 A8 AF 20 49 42	4D 20 50 43 3A 20 24 82	\$Тип IBM PC: \$B
0000000280:	A5 E0 E1 A8 EF 20 8E 91	3A 20 24 3C 32 0D 0A 24	ерсия ОС: \$<2\$
0000000290:	4F 45 4D 3A 20 24 91 A5	E0 A8 A9 AD EB A9 20 AD	OEM: \$Серийный н
00000002A0:	AE AC A5 E0 20 AF AE AB	EC A7 AE A2 A0 E2 A5 AB	омер пользовател
00000002B0:	EF 3A 20 24 00 00 00 00	00 00 00 00 00 00 00 00	я: \$
00000002C0:	24 0F 3C 09 76 02 04 07	04 30 CB 51 8A E0 9A 00	\$<ov0♦♦0TKpb
00000002D0:	00 0C 00 86 C4 B1 04 D2	E8 9A 00 00 0C 00 59 CB	♀ Ж-♦шб ♀ Y
00000002E0:	53 8A FC 9A 0B 00 0C 00	88 25 4F 88 05 4F 8A C7	SKMъ♀ И%ОИ+ОК
00000002F0:	9A 0B 00 0C 00 88 25 4F	88 05 5B CB 51 52 32 E4	б♂ ♀ И%ОИ+ [QR2ф
0000000300:	33 D2 B9 0A 00 F7 F1 80	CA 30 88 14 4E 33 D2 3D	зтт\$ yèA=0ИГНЗт=
0000000310:	0A 00 73 F1 3C 00 74 04	0C 30 88 04 5A 59 CB 50	sèc t♀0И+ZYтP
0000000320:	06 B8 00 F0 8E C0 26 A0	FE FF 3C FF 74 1C 3C FE	♀j ÈO L&a■ < tL<■
0000000330:	74 1E 3C FD 74 20 3C FC	74 22 3C FB 74 12 3C FA	t▲<xt <N"t"<vt\$<·
0000000340:	74 20 3C F9 74 28 3C F8	74 1E BA 08 00 EB 22 90	t <·t(<°t▲■ б"P
0000000350:	BA 0D 00 EB 1C 90 BA 3A	00 EB 16 90 BA 15 00 EB	бLP  : бP  \$ б
0000000360:	10 90 BA 1A 00 EB 0A 90	BA 2A 00 EB 04 90 BA 41	►P  → бP  * бP  A
0000000370:	00 07 58 CB 3C 00 74 1B	BE 00 00 9A 3C 00 0C 00	·Xт< t← б< ♀
0000000380:	B2 2E 83 C6 02 88 14 46	8A C4 9A 3C 00 0C 00 BA	■.ГтИГFK-б< ♀
0000000390:	00 00 CB BA 6B 00 CB 8A	C7 BE 00 00 83 C6 02 9A	т  k тK  д Гтб
00000003A0:	3C 00 0C 00 BA 00 00 CB	8A C3 BF 00 00 9A 0B 00	< ♀    тK   б♂
00000003B0:	0C 00 88 25 88 45 01 83	C7 05 8B C1 9A 20 00 0C	♀ И%ИЕОГ  +л-б ♀
00000003C0:	00 BA 00 00 CB 50 B4 09	CD 21 58 CB 2B C0 50 8C	тP  o=!Xт+ LPM
00000003D0:	D8 50 B8 02 00 8E D8 B8	00 00 BA 52 00 9A 05 01	тPт 0 от   R б+0
00000003E0:	0C 00 9A 5F 00 0C 00 9A	05 01 0C 00 B4 30 CD 21	♀ б_ ♀ б+0♀ т0=!
00000003F0:	BA 5F 00 9A 05 01 0C 00	9A B4 00 0C 00 9A 05 01	б+0♀ б- ♀ б+0
0000000400:	0C 00 BA 70 00 9A 05 01	0C 00 9A D7 00 0C 00 9A	♀   p б+0♀ б- ♀ б
0000000410:	05 01 0C 00 BA 76 00 9A	05 01 0C 00 9A E8 00 0C	+0♀   v б+0♀ бш ♀
0000000420:	00 9A 05 01 0C 00 32 C0	B4 4C CD 21 CB	б+0♀ 2 L=!

Рисунок 8 — Структура «хорошего» EXE файла, адреса 0h-240h

## Загрузка COM модуля в основную память

Загруженный в основную память под отладчиком TD.EXE COM модуль показан на рис. 9.

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

В выделенный системой фрагмент памяти сначала загружается PSP, а начало кода определяется директивой ORG от начала выделенного фрагмента. В данном случае код располагается с адреса 48DD:0100.

2. Что располагается с адреса 0?

С адреса 0 располагается PSP.

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

Все сегментные регистры имеют значение 48DD и указывают на PSP.

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

Стек занимает весь фрагмент памяти, выделенный под программу и определяется регистрами SS и SP. Он занимает адреса 0000h-FFFEh.

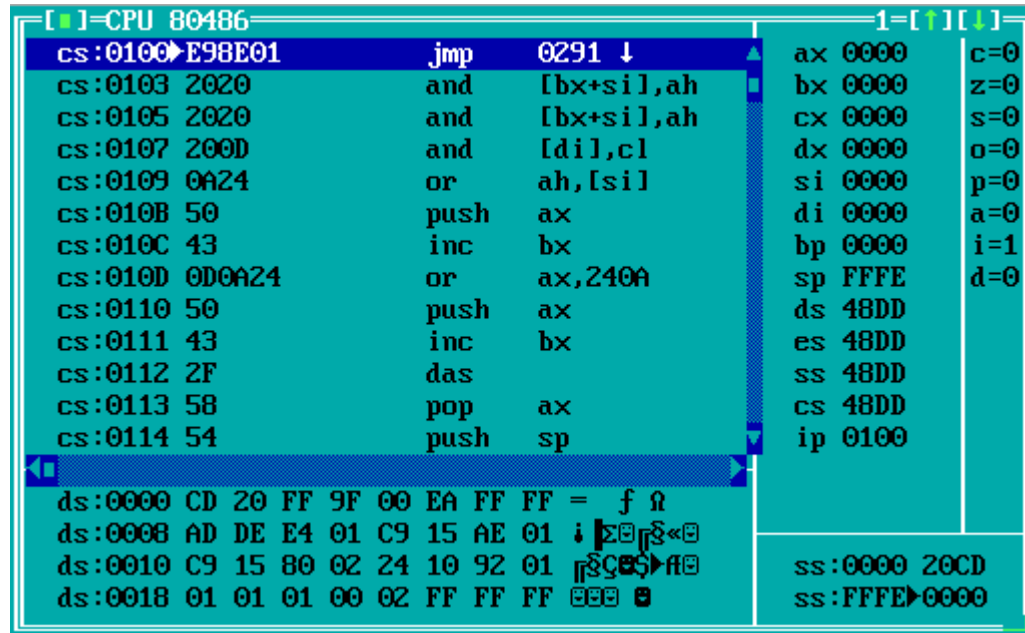


Рисунок 9 — COM модуль в отладчике TD.EXE

### Загрузка «хорошего» EXE модуля в основную память

Загруженный в основную память под отладчиком TD.EXE «хороший» EXE модуль показан на рис. 10.

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

Сначала помещается PSP, а далее устанавливаются сегментные регистры. CS=48F9 – начало сегмента кода, SS=48ED – начало сегмента стека, DS=ES=48DD – начало фрагмента, выделенного программе.

2. На что указывают регистры DS и ES?

На PSP.

3. Как определяется стек?

Для стека в программе выделяется отдельный сегмент с параметром STACK. SS указывает на начало стека, а SP – на верхушку стека.

4. Как определяется точка входа?

Точка входа определяется с помощью директивы END <точка\_входа>.



Точкой входа можно указать как процедуру, так и метку.

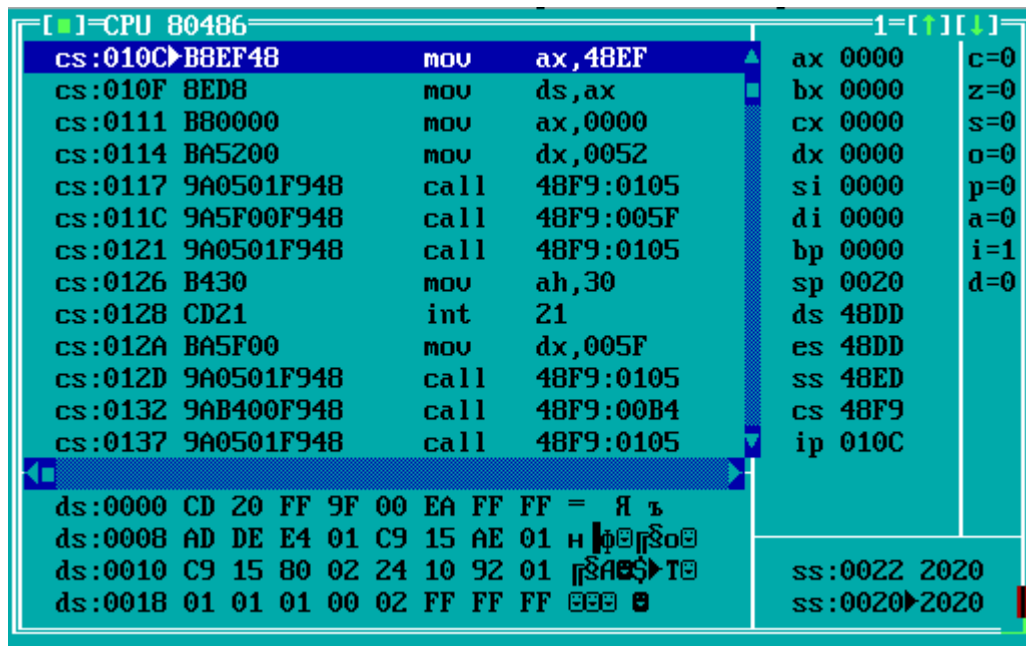


Рисунок 10 — «Хороший» EXE модуль в отладчике TD.EXE

### Выводы:

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