**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №7**

**по дисциплине «Операционные системы»**

Тема: Построение модуля оверлейной структуры

|  |  |  |
| --- | --- | --- |
| Студент гр. 6383 |  | Яковлев Е.А. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

**Цель лабораторной работы**

Исследование возможности построения загрузочного модуля оверлейной структуры. Исследуется структура оверлейного сегмента и способ загрузки и выполнения оверлейных сегментов. Для запуска вызываемого оверлейного модуля используется функция 4B03h прерывания int 21h. Все загрузочные и оверлейные модули находятся в одном каталоге.

**Постановка задачи**

Шаг 1. Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .ЕХЕ, который выполняет функции:

1) Освобождает память для загрузки оверлеев.

2) Читает размер файла оверлея и запрашивает объем памяти, достаточный для его загрузки.

3) Файл оверлейного сегмента загружается и выполняется.

4) Освобождается память, отведенная для оверлейного сегмента.

5) Затем действия 1)-4) выполняются для следующего оверлейного сегмента.

Шаг 2. Также необходимо написать и отладить оверлейные сегменты. Оверлейный сегмент выводит адрес сегмента, в который он загружен.

Шаг 3. Запустите отлаженную программу. Оверлейные сегменты должны загружаться с одного и того же адреса, перекрывая друг друга.

Шаг 4. Запустите приложение из другого каталога. Приложение должно быть выполнено успешно.

Шаг 5. Запустите приложение, когда одного оверлея нет в каталоге. Приложение должно закончиться аварийно.

Шаг 6. Занесите полученные результаты в виде скриншотов в отчёт. Оформите отчёт в соответствии с требованиями

Процедуры, которые используются в программе.

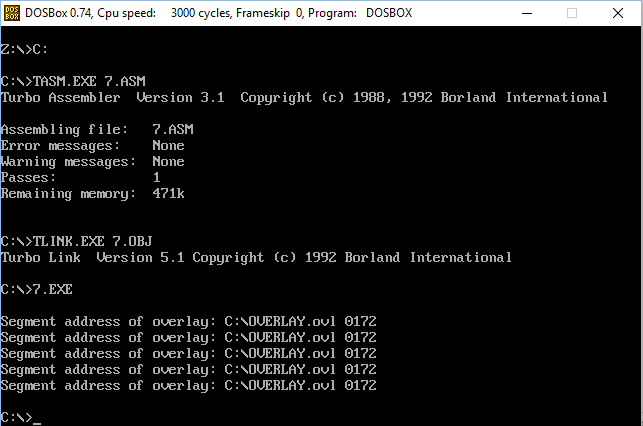
|  |  |
| --- | --- |
| writeString | Вывод строки на экран. |
| fileNameTest | Запоминание имени оверлея. |
| clearMem | Освобождение памяти перед загрузкой оверлея. |
| memForDTA | Определение области памяти под буфер DTA. |
| downloadProgram | Загрузка программы в отведенную область памяти. |
| clearOverlayMem | Очистка памяти после отработки оверлея. |
| fileNotFound | Вывод информации о том, что файл не найден. |
| doOverlay | Выполнение оверлея. |

Переменные, которые используются в программе.

|  |  |  |
| --- | --- | --- |
| DTA | db | Буфер DTA. |
| TEMP\_SS | db | Запоминание сегмента SS. |
| TEMP\_SP | db | Запоминание сегмента SP. |
| SEGADD | db | Вывод информации о адресе сегмента оверлея. |
| NO\_FILE | db | Вывод информации о том, что файл не найден. |
| NO\_MEMORY | dw | Вывод информации о том, что мало памяти. |
| parameters | dw | Запоминание расположения оверлея. |

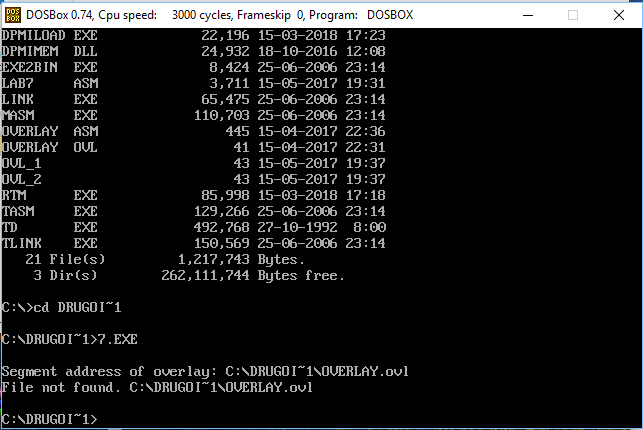
**Ход выполнения работы**

*Шаг 1:* Результаты выполнения работы представлены на рисунке №1:



*Рис. 1 – результат работы программы, находящейся с оверлеями в одной директории*

*Шаг 2:* Результаты выполнения работы представлены на рисунке №2:

**

*Рис. 2 – результат работы программы, находящейся с оверлеями в разных директориях*

*Шаг 3:* Результаты выполнения работы представлены на рисунке №3:

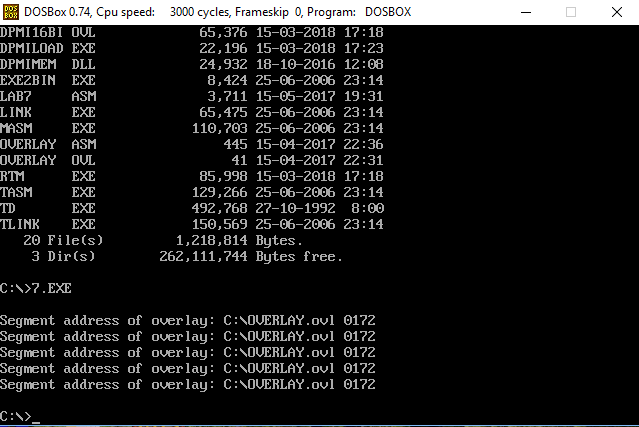


Рис. 3 - *результат работы программы, находящейся с одним оверлеем в директории*

**Контрольные вопросы**

1. *Как должна быть устроена программа, если в качестве оверлейного сегмента использовать COM-модули?*

В COM-модуле после записи значений регистров в стек, необходимо поместить значение регистра CS в регистр DS, так как адрес сегмента данных совпадает с адресом сегмента кода и и в первые 256 байт записать содержимое PSP вызывающей программы так как PSP запускаемой программы при таком вызове сформирован не был бы.

**Вывод**

В результате выполнения данной лабораторной работы были исследованы организация загрузочных модулей оверлейной структуры. Была написана программа, в которой ошибок не обнаружено.