

**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**

Студентка гр. 9383

\_\_\_\_\_

Лихашва А.Д.

Преподаватель

\_\_\_\_\_

Ефремов М.А.

Санкт-Петербург

2021

## **Постановка задачи**

### **Цель 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.

В этой работе также рассматривается приложение, состоящее из нескольких модулей, поэтому все модули помещаются в один каталог и вызываются с использованием полного пути.

### **Задание.**

**Шаг 1.** Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .EXE, 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 отлаженное приложение. 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 в случае, когда одного оверлея нет в каталоге. Приложение должно закончиться аварийно.

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

### **Выполнение шагов лабораторной работы:**

#### **Шаг 1.**

Был написан и отлажен программный модуль типа .EXE, который выполняет функции:

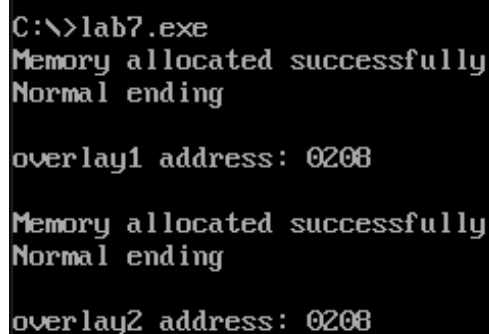
- 1) Освобождает память для загрузки оверлеев.
- 2) Читает размер файла оверлея и запрашивает объем памяти, достаточный для его загрузки.
- 3) Файл оверлейного сегмента загружается и выполняется.
- 4) Освобождается память, отведенная для оверлейного сегмента.
- 5) Затем действия 1)-4) выполняются для следующего оверлейного сегмента.

#### **Шаг 2.**

Были написаны и отлажены оверлейные сегменты. Оверлейный сегмент выводит адрес сегмента, в который он загружен.

#### **Шаг 3.**

Было запущено отлаженное приложение. Оверлейные сегменты загружаются с одного и того же адреса, перекрывая друг друга.



```
C:\>lab7.exe
Memory allocated successfully
Normal ending

overlay1 address: 0208

Memory allocated successfully
Normal ending

overlay2 address: 0208
```

Рисунок 1: Работа lab7.exe

#### Шаг 4.

Было запущено приложение из другого каталога

```
Memory allocated successfully
Normal ending

overlay1 address: 0208

Memory allocated successfully
Normal ending

overlay2 address: 0208
```

Рисунок 2: Работа lab7.exe из другого каталога

#### Шаг 5.

Было запущено приложение в случае, когда одного оверлея нет в каталоге.

```
Memory allocated successfully
Normal ending

overlay1 address: 0208

File not found
Loading error: file was not found
```

Рисунок 3: Работа lab7.exe без первого оверлея

### Ответы на контрольные вопросы

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

Если в качестве оверлейного сегмента использовать .COM модули, то при обращении к оверлейному сегменту необходимо учитывать смещение 100h, потому что в .COM модуле присутствует PSP.

### **Заключение.**

Был написан загрузочный модуль оверлейной структуры, а также написаны сами оверлеи. Изучены дополнительные функции работы с памятью и способы загрузки.