

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по практической работе №7**  
**по дисциплине «Операционные системы»**  
**Тема: Построение модуля оверлейной структуры**

Студент гр. 8382

Торосян Т.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

### **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 .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 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

## Структуры данных.

Таблица 1 - Структуры данных используемые в программе

Название	Тип	Назначение
DTA_MEM	db	Буфер DTA.
ALLOCATION_END	db	Вывод информации о успешности аллокации
OVLS_ADDR	db	Вывод информации о адресе сегмента оверлея.
ALL_FILE_ERR	db	Вывод информации о том, что файл не найден.
ALL_ROUTE_ERR	dw	Вывод информации о том, что маршрут не найден

## Результат работы.

- 1) Результаты выполнения работы программы, при условии что оверлей файлы находятся в той же директории, представлены на рисунке 1.

```
C:\>lab7
memory has been freed
Allocation was successful
Load was successful

FILE1_OVL address:0207

Allocation was successful
Load was successful

FILE2_OVL address:0207
C:\>_
```

Рисунок 1 – С оверлеями

- 2) Результаты выполнения работы программы, при условии запуска ее из другой директории, представлены на рисунке 2.

```
C:\>COMPILED\lab7
memory has been freed
Allocation was successful
Load was successful

FILE1_OVL address:0207

Allocation was successful
Load was successful

FILE2_OVL address:0207
C:\>_
```

Рисунок 2 – Из другой директории

3) Результаты выполнения работы программы, при условии что 1-ый оверлей находится в другой директории, представлены на рисунке 3.

```
C:\>COMPILED\lab7
memory has been freed
ERR: file not found(allocation err)
ERR: file not found(load err)

Allocation was successful
Load was successful

FILE2_OVL address:0207
C:\>
```

Рисунок 3 – С одним оверлеем

4) Результаты выполнения работы программы, при условии что два оверлея находятся в другой директории, представлены на рисунке 4.

```
C:\>COMPILED\lab7
memory has been freed
ERR: file not found(allocation err)
ERR: file not found(load err)

ERR: file not found(allocation err)
ERR: file not found(load err)

C:\>_
```

Рисунок 4 - Без оверлеев

### Ответы на 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, кроме того необходимо добавить 100h, т. к. изначально данные сегменты настроены на PSP.

### Вывод.

В результате выполнения данной лабораторной работы были исследована организация загрузочных модулей оверлейной структуры. Была написана программа, не содержащая критических ошибок.