

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. 9382

Голубева В.П.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

Исследование возможности построения загрузочного модуля оверлейной структуры. Исследуется структура оверлейного сегмента и способ загрузки и выполнения оверлейных сегментов.

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 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 работы.

Запустим программу из директории с оверлеями. Результат можно посмотреть в Рисунке 1.

```
C:\>LAB7.EXE
memory has been freed
allocation_mem was successful
load was successful

file1_ovl address:0209

allocation_mem was successful
load was successful

file2_ovl address:0209
```

Рисунок 1. Запуск программы из директории с оверлеями

Теперь запустим программу, находясь в другой директории.

```
C:\>LR7\LAB7.EXE
memory has been freed
allocation_mem was successful
load was successful

file1_ovl address:0209

allocation_mem was successful
load was successful

file2_ovl address:0209
```

Рисунок 2. Запуск программы во время нахождения в другой директории

Теперь запустим программу при условии, что первый оверлей находится в другой директории.

```
C:\>LAB7.EXE
memory has been freed
error: file not found(allocation_mem err)
error: file not found(load err)

allocation_mem was successful
load was successful

file2_ovl address:0209
```

Рисунок 3. Запуск программы при условии, что первый оверлей находится в другой директории

Теперь запустим программу при условии, что второй оверлей находится в другой директории.

```
C:\>LAB7.EXE
memory has been freed
allocation_mem was successful
load was successful

file1_ovl address:0209

error: file not found(allocation_mem err)
error: file not found(load err)
```

Рисунок 4. Запуск программы при условии, что второй оверлей находится в другой директории

Теперь запустим программу при условии, что оба оверлея находятся в другой директории.

```
C:\>LR7\LAB7.EXE
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)
```

Рисунок 5. Запуск программы при условии, что оба оверлея находятся в другой директории

Ответы на контрольные вопросы.

Контрольные вопросы по лабораторной работе №7

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

Ответ: 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.

Была исследована структура, способ загрузки и выполнение оверлейных сегментов.