

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

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

Студентка гр. 8382

Кулачкова М.К.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

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.

Ход выполнения работы.

Была реализована программа, которая загружает и выполняет оверлейные сегменты. Так как при первоначальном запуске программы ей отводится вся доступная память, для загрузки оверлеев осуществляется освобождение памяти, не используемой программой, при помощи функции 4Ah прерывания int 21h. При возникновении ошибки при освобождении памяти, выводится соответствующее сообщение и осуществляется выход из программы.

Затем формируется путь до оверлея. Для этого в специально выделенную строку записывается путь до вызываемой программы, который расположен после переменных среды, после чего имя вызываемого модуля заменяется именем загружаемого оверлея.

После этого определяется размер оверлея при помощи функции 4Eh прерывания int 21h. Перед обращением к функции 1Ah прерывания int 21h устанавливается адрес для буфера DTA, под который в сегменте данных предварительно была выделена область памяти размеров в 43 байта. В регистре CX устанавливается атрибут, который для файла имеет значение 0. После запуска функции проверяется значение флага переноса CF и в случае ошибки выводится соответствующее сообщение и осуществляется выход из программы. Если функция была выполнена успешно, то в области памяти буфера DTA со смещением 1Ah будет находиться младшее слово размера файла в байтах, а со

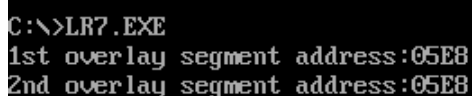
смещением 1Ch – старшее. Размер файла переводится в параграфы. Затем с помощью функции 48h прерывания int 21h под оверлей выделяется блок памяти. Сегментный адрес выделенного блока сохраняется в переменную.

Для загрузки оверлея используется функция 4B03h прерывания int 21h. Перед вызовом функции ей передаются параметры: в DS:DX – указатель на строку, содержащую путь к оверлею, в ES:BX – указатель на сегментный адрес загрузки программы. Если при выполнении функции произошла ошибка, выводится сообщение, соответствующее коду ошибки, и осуществляется выход из программы. Если ошибки не было, оверлей загружен в память. Он вызывается по своему сегментному адресу. После вызова оверлея происходит освобождение занимаемой им памяти.

Программа выполняет перечисленные действия для двух оверлейных сегментов.

Оверлейный сегмент представляет собой кодовый сегмент, который оформлен как функция и выводит на экран адрес сегмента, в который он загружен.

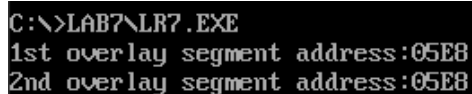
Полученная программа была запущена. Результат ее работы представлен на рисунке 1. Можно заметить, что сегменты загружаются с одного и того же адреса, перекрывая друг друга.



```
C:\>LR7.EXE
1st overlay segment address:05E8
2nd overlay segment address:05E8
```

Рисунок 1 - Результат запуска приложения

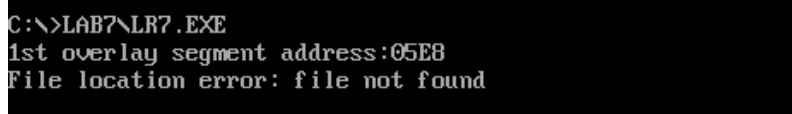
Приложение было запущено из другого каталога и было выполнено успешно. Результат работы представлен на рисунке 2.



```
C:\>LAB7\LR7.EXE
1st overlay segment address:05E8
2nd overlay segment address:05E8
```

Рисунок 2 - Результат запуска приложения из другого каталога

Из каталога был удален второй оверлей. При запуске приложения был выполнен первый оверлей, при попытке загрузки второго программа аварийно завершилась. Результат ее работы представлен на рисунке 3.



```
C:\>LAB7\LR7.EXE
1st overlay segment address:05E8
File location error: file not found
```

Рисунок 3 - Результат запуска приложения после удаления одного из оверлеев

Ответы на вопросы.

1. Как должна быть устроена программа, если в качестве оверлейного сегмента использовать .COM модули?
– Оверлейный сегмент является не загрузочным модулем, а кодовым сегментом, оформленным как функция с точкой входа по адресу 0, поэтому при использовании модуля .COM в качестве оверлейного сегмента необходимо учитывать, что кодовый сегмент модуля смещен на 100h байт, занимаемых PSP, относительно его начала.

Выводы.

Было реализовано приложение, состоящее из вызывающей программы и двух оверлейных сегментов, расположенных в одном каталоге. В ходе выполнения работы исследована структура оверлейного сегмента и способ загрузки и выполнения оверлейных сегментов.