

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

ОТЧЕТ
по лабораторной работе №6
по дисциплине «Операционные системы»
Тема: Построение модуля динамической структуры.

Студент гр. 0381

Самойлов З. А.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

Цель работы.

Исследование возможности построения загрузочного модуля динамической структуры. В отличие от предыдущих лабораторных работ в этой работе рассматривается приложение, состоящее из нескольких модулей, а не из одного модуля простой структуры. В этом случае разумно предположить, что все модули приложения находятся в одном каталоге и полный путь в этот каталог можно взять из среды, как это делалось в работе 2. Понятно, что такое приложение должно запускаться в соответствии со стандартами ОС. В работе исследуется интерфейс между вызывающим и вызываемым модулями по управлению и по данным. Для запуска вызываемого модуля используется функция 4B00h прерывания int 21h. Все загрузочные модули находятся в одном каталоге. Необходимо обеспечить возможность запуска модуля динамической структуры из любого каталога.

Постановка задачи

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

- 1) Подготавливает параметры для запуска загрузочного модуля из того же каталога, в котором находится он сам. Вызываемому модулю передается новая среда, созданная вызывающим модулем и новая командная строка.
- 2) Вызываемый модуль запускается с использованием загрузчика.
- 3) После запуска проверяется выполнение загрузчика, а затем результат выполнения вызываемой программы. Необходимо проверять причину завершения и, в зависимости от значения, выводить соответствующее сообщение. Если причина завершения 0, то выводится код завершения.

В качестве вызываемой программы необходимо взять программу ЛР 2, которая распечатывает среду и командную строку. Эту программу следует немного модифицировать, вставив перед выходом из нее обращение к функции ввода символа с клавиатуры. Введенное значение записывается в регистр AL и затем происходит обращение к функции выхода 4Ch прерывания int 21h.

Шаг 2. Запустите отлаженную программу, когда текущим каталогом является каталог с разработанными модулями. Программа вызывает другую программу, которая останавливается, ожидая символ с клавиатуры. Введите произвольный символ из числа A-Z. Посмотрите причину завершения и код. Занесите полученные данные в отчет.

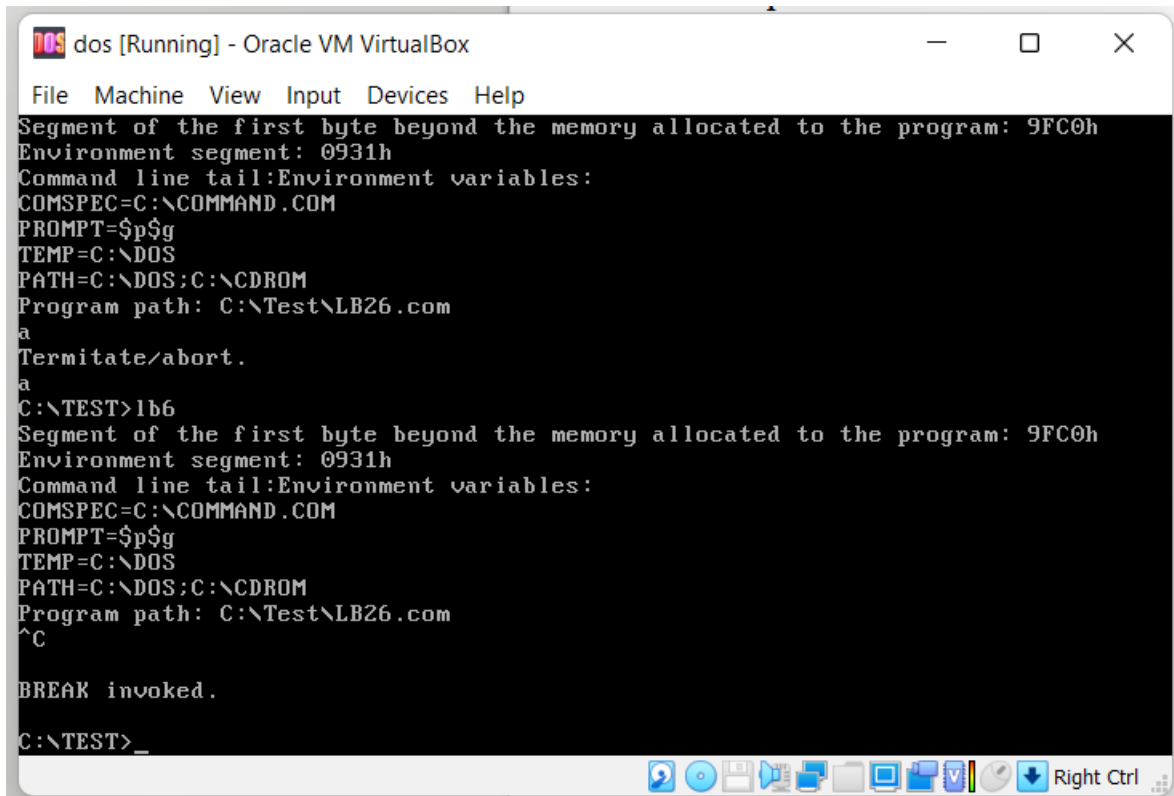
Шаг 3. Запустите отлаженную программу, когда текущим каталогом является каталог с разработанными модулями. Программа вызывает другую программу, которая останавливается, ожидая символ с клавиатуры. Введите комбинацию символов Ctrl-C. Посмотрите причину завершения и код. Занесите полученные данные в отчет.

Шаг 4. Запустите отлаженную программу, когда текущим каталогом является какойлибо другой каталог, отличный от того, в котором содержатся разработанные программные модули. Повторите ввод комбинаций клавиш. Занесите полученные данные в отчет.

Шаг 5. Запустите отлаженную программу, когда модули находятся в разных каталогах. Занесите полученные данные в отчет.

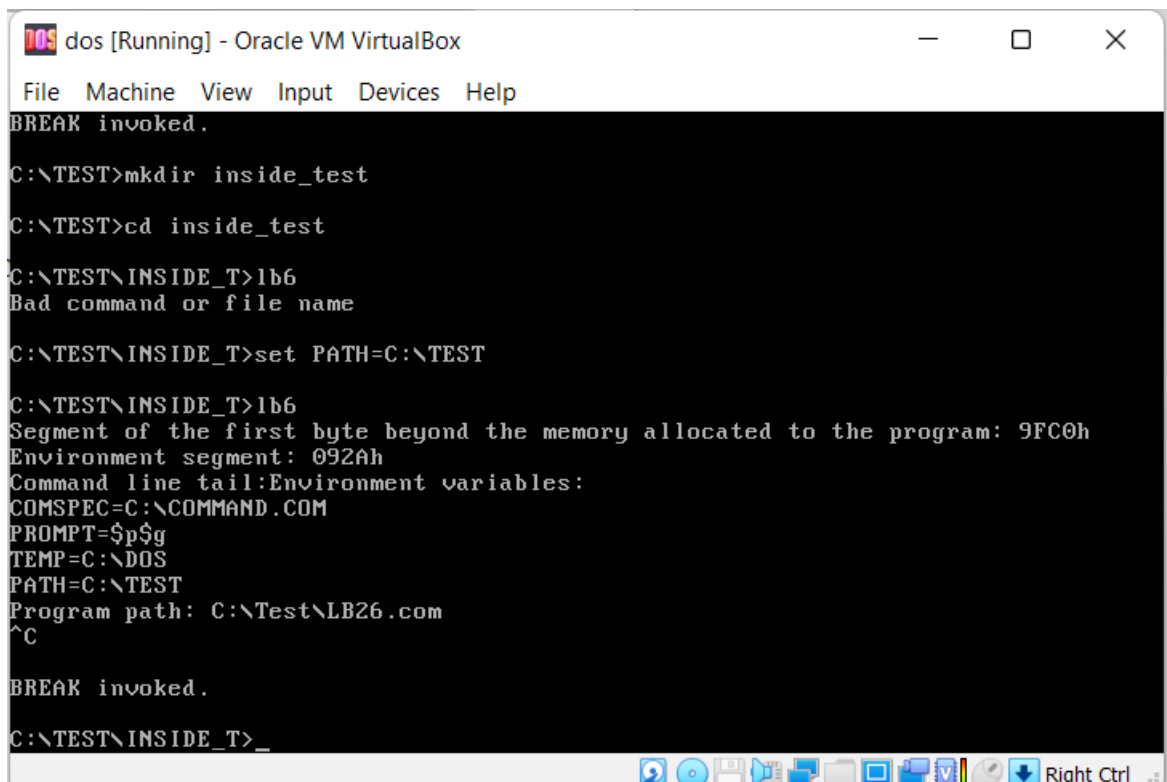
Выполнение работы.

При выполнении первого, второго и третьего шагов была написана и запущена программа.



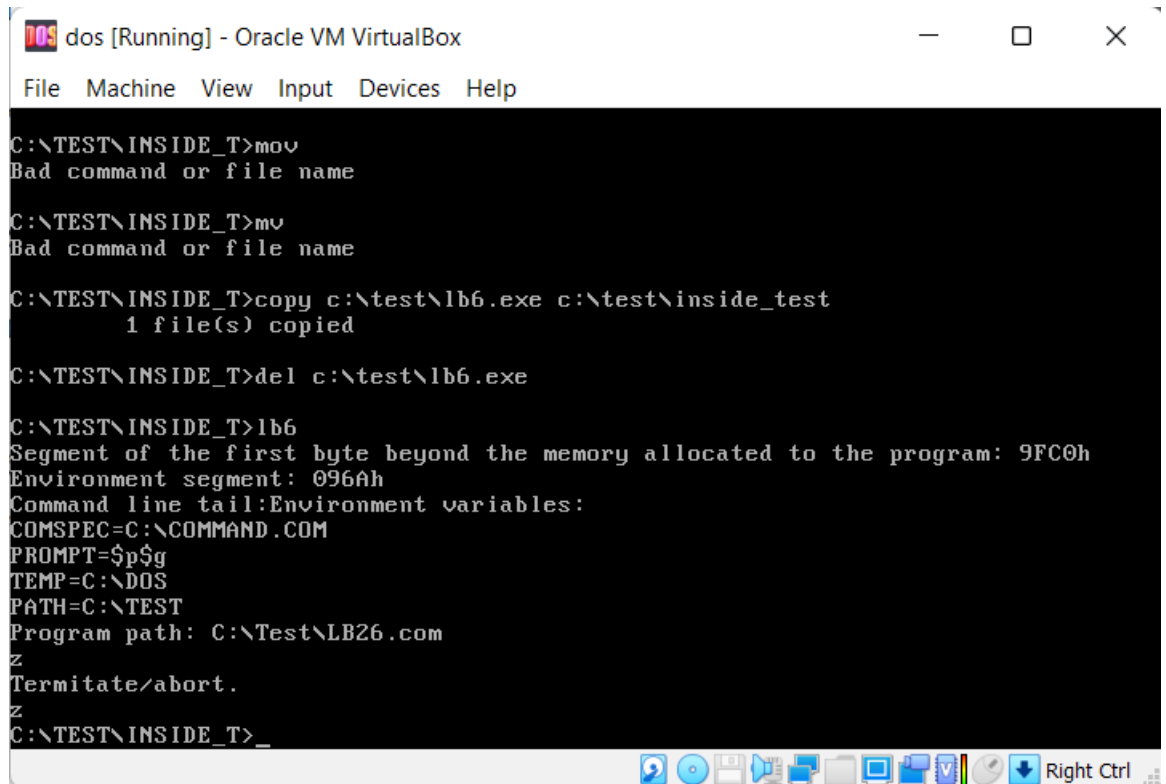
Скриншот 1. Результат выполнения шагов 2 и 3.

Для шага 4 в переменную PATH среды был добавлен путь к папке с lb26 и lb6.



Скриншот 2. Результат выполнения шага 4.

Шаг 5.



```
dos [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

C:\TEST\INSIDE_T>mov
Bad command or file name

C:\TEST\INSIDE_T>mv
Bad command or file name

C:\TEST\INSIDE_T>copy c:\test\lb6.exe c:\test\inside_test
1 file(s) copied

C:\TEST\INSIDE_T>del c:\test\lb6.exe

C:\TEST\INSIDE_T>lb6
Segment of the first byte beyond the memory allocated to the program: 9FC0h
Environment segment: 096Ah
Command line tail:Environment variables:
COMSPEC=C:\COMMAND.COM
PROMPT=$p$g
TEMP=C:\DOS
PATH=C:\TEST
Program path: C:\Test\LB26.com
Z
Terminate/abort.
Z
C:\TEST\INSIDE_T>
```

Скриншот 3. Результат выполнения шага 5.

Вывод.

Был построен пользовательский обработчик прерывания сигналов клавиатуры, использующий оригинальный обработчик.

Ответы на контрольные вопросы.

1) Как реализовано прерывание CTRL-C?

Ответ. При обнаружении комбинации CTRL-C или CTRL-BREAK вызывается прерывание `int 23h`, которое завершает выполнение программы. Поведение прерывания можно изменить, заменив его пользовательским.

2) В какой точке заканчивается вызываемая программа, если код причины завершения 0?

Ответ. Выполнение вызываемой программы завершается вызовом прерывания `int 21h – AH = 4Ch` в конце процедуры `Main`.

2) В какой точке заканчивается вызываемая программа по прерыванию `Ctrl-C`?

Ответ. Выполнение вызываемой программы завершается при выполнении прерывания `int 21h – AH = 01h`, так как вызывается прерывание `int 23h`.