

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

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

Студентка гр. 8381

Лисок М.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

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.

Описание функций и структур данных.

Таблица 1 – функции управляющей программы.

Название функции	Назначение
PRINT	Печатает строку на экран.
FreeMemory	Освобождает память, занимаемую вызывающей программой, и создает блок параметров для вызываемой программы.
BYTE_TO_HEX	Переводит число AL в коды символов 16 с/с, записывая получившиеся в AL и AH.
TETR_TO_HEX	Вспомогательная функция для работы BYTE_TO_HEX
MAIN	Основная функция программы.
RunProc	Функция запуска вызываемого модуля

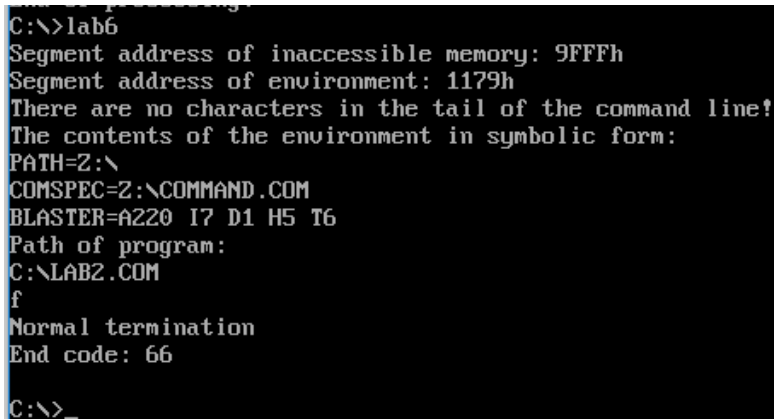
Ход работы.

Программный .EXE модуль, выполняет следующие функции:

- 1.Подготавливает параметры для запуска загрузочного модуля из того же каталога, в котором он находится сам. Вызываемому модулю передаёт новую среду и новую командную строку.
- 2.Вызываемый модуль запускается с использованием загрузчика.

После запуска проверяется выполнение загрузчика, а затем результат выполнения вызываемой программы. Проверяет причину завершения и, в зависимости от значения, выводит соответствующее сообщение. Если причина завершения 0, то выводится код завершения.

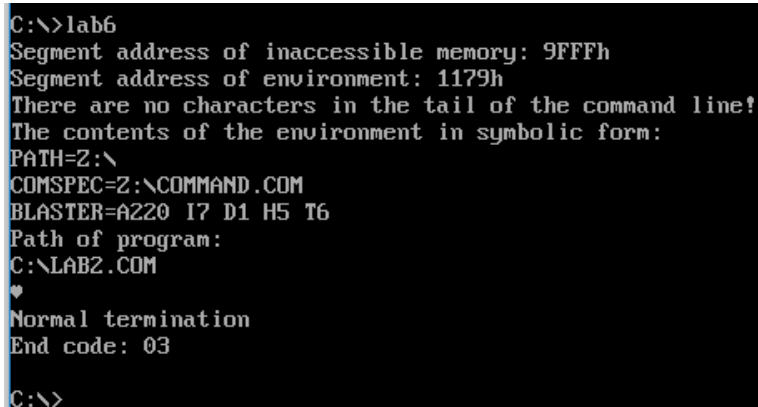
Была запущена программа, когда оба модуля находятся в текущем каталоге и введен символ 'f'. Результат работы программы представлен на рис. 1.



```
C:\>lab6
Segment address of inaccessible memory: 9FFFh
Segment address of environment: 1179h
There are no characters in the tail of the command line!
The contents of the environment in symbolic form:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of program:
C:\LAB2.COM
f
Normal termination
End code: 66
C:\>_
```

Рисунок 1 – результат работы программы, оба модуля в текущем каталоге.

Была запущена программу, когда оба модуля находятся в текущем каталоге и введена комбинация клавиш “Ctrl+C”. Результат работы программы представлен на рис. 2.



```
C:\>lab6
Segment address of inaccessible memory: 9FFFh
Segment address of environment: 1179h
There are no characters in the tail of the command line!
The contents of the environment in symbolic form:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of program:
C:\LAB2.COM
^
Normal termination
End code: 03
C:\>
```

Рисунок 2 – результат работы программы, оба модуля в текущем каталоге, ввод комбинации клавиш “Ctrl+C”.

Была запущена программа, когда оба модуля находятся не в текущем каталоге, и введен символ ‘k’. Результат работы программы представлен на рис. 3.

```
C:\NEWDIR>lab6
Segment address of inaccessible memory: 9FFFh
Segment address of environment: 1191h
There are no characters in the tail of the command line!
The contents of the environment in symbolic form:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of program:
C:\NEWDIR\LAB2.COM
k
Normal termination
End code: 6B
C:\NEWDIR>
```

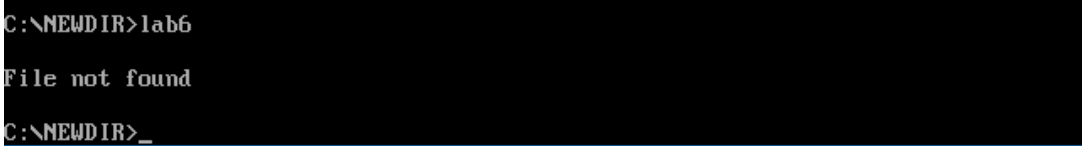
Рисунок 3 – результат работы программы, оба модуля не в текущем каталоге, ввод символа ‘k’.

Была запущена программа, когда оба модуля находятся не в текущем каталоге и введена комбинация клавиш “Ctrl+C”. Результат работы программы представлен на рис. 4.

```
C:\NEWDIR>lab6
Segment address of inaccessible memory: 9FFFh
Segment address of environment: 1191h
There are no characters in the tail of the command line!
The contents of the environment in symbolic form:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of program:
C:\NEWDIR\LAB2.COM
♥
Normal termination
End code: 03
C:\NEWDIR>_
```

Рисунок 4 – результат работы программы, оба модуля не в текущем каталоге, ввод комбинации клавиш “Ctrl+C”.

Была запущена программа, когда модули находятся в разных каталогах. Результат работы программы представлен на рис. 5.



```
C:\NEWDIR>lab6
File not found
C:\NEWDIR>_
```

Рисунок 5 – результат работы программы, модули находятся в разных каталогах.

Выводы.

В процессе выполнения данной лабораторной работы была исследована возможность построения загрузочного модуля динамической структуры.

Ответы на контрольные вопросы.

1. Как реализовано прерывание Cntrl-C?

Ответ: При нажатии комбинации клавиш Cntrl-C вызывается прерывание 23h, которое завершает текущий процесс и передает управление порождаемому процессу.

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

Ответ: В точке вызова функции 4Ch прерывания int 21h.

3. В какой точке заканчивается программа по прерыванию Cntrl-C?

Ответ: В точке вызова функции 01h прерывания int 21h.