

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.

Студентка гр. 7381

Процветкина А.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2019

Цель работы.

Исследование возможности построения загрузочного модуля динамической структуры. В отличии 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.

Функции и структуры данных управляющей программы.

Таблица 1 – Функции управляющей программы

Название функции	Назначение
TETR_TO_HEX	Функция шаблона, приведенного в методических указаниях. Функция переводит половину байта в шестнадцатеричную систему.
BYTE_TO_HEX	Функция шаблона, приведенного в методических указаниях. Байт в регистре AL переводится в два символа шестнадцатеричного числа в регистре AX.
PRINT	Функция выводит сообщение на экран.
ERROR_PROCESSING	Функция обработки ошибок при вызове функции 4Ah прерывания int 21h для освобождения места в памяти.

CLEAR_MEMORY	Функция освобождения памяти. В случае возникновения ошибок при освобождении памяти вызывается функция ERROR_PROCESSING.
CREATION_PARAMETER_BLOCK	Функция создает блок параметров.
ERR_PROCESSING	Функция обработки ошибок, если вызываемая программа не была загружена.
COMPLETION_PROCESSING	Функция обработки причины завершения.
BASE_PROCESS	Функция для запуска дочерней программы.
MAIN PROC	Основная функция.

Таблица 2 – Структура данных управляющей программы

Имя	Тип	Назначение
parameter_block	dw dd dd dd	сегментный адрес среды сегмент и смещение командной строки сегмент и смещение первого FCB сегмент и смещение второго FCB
error1_7	db	Вывод строки 'Memory control block destroyed'
error1_8	db	Вывод строки 'Not enough memory to perform the function'
error1_9	db	Вывод строки 'Wrong memory address'
error2_1	db	Вывод строки 'Number of function is incorrect'
error2_2	db	Вывод строки 'File not found'
error2_5	db	Вывод строки 'Disk error'
error2_8	db	Вывод строки 'Insufficient memory'
error2_10	db	Вывод строки 'Incorrect environment string'

error2_11	db	Вывод строки 'Wrong format'
end0	db	Вывод строки 'Normal completion'
end1	db	Вывод строки 'Completion by Ctrl-Break'
end2	db	Вывод строки 'Completion by device error'
end3	db	Вывод строки 'Completion by function 31h'
output_code	db	Вывод строки 'End code: '
keep_ss	dw	Переменная для сохранения значения регистра SS.
keep_sp	dw	Переменная для сохранения значения регистра SP.

Выполнение работы.

Был написан программный модуль типа .EXE, который выполняет следующие 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.

Был выполнен запуск отлаженной программы, когда текущим каталогом является каталог с разработанными модулями. Для проверки работы программы был введен случайный символ A-Z.

```

C:\>LAB6.EXE
Unaccessable memory starts from: 9FFF
Segment adress provided to the program: 01E5
Tail of the command line:
Enviroment content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of the program: C:\LAB2.COM
a
Normal completion
End code: 61

```

Рисунок 1 – Запуск программы лабораторной работы №6

Проверить работоспособность программы используя сочетание CTRL-C в DOS не представляется возможным.

Был выполнен запуск отлаженной программы, когда текущим каталогом является какой-либо другой каталог, отличный от того, в котором содержатся разработанные программные модули.

```

C:\>TASM\LAB6.EXE
Unaccessable memory starts from: 9FFF
Segment adress provided to the program: 01E5
Tail of the command line:
Enviroment content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path of the program: C:\TASM\LAB2.COM
S
Normal completion
End code: 53

```

Рисунок 2 – Запуск программы лабораторной работы №6.

Был выполнен запуск отлаженной программы, когда программные модули находятся в разных каталогах.

```

C:\>TASM\LAB6.EXE
File not found

```

Рисунок 3 – Запуск программы лабораторной работы №6.

В результате выполнения программы на экран было выведено сообщение об ошибке.

Ответы на контрольные вопросы.

Вопрос: Как реализовано прерывание Ctrl-C?

Ответ: При нажатии комбинации клавиш Ctrl-C вызывается прерывание 23h. Адрес в этом векторе (0000:008c) – адрес, по которому передается управление. Обычная системная обработка Ctrl-C сводится к немедленному снятию программы.

Вопрос: В какой точке заканчивается вызываемая программа, если код причины завершения 0?

Ответ: Вызываемая программа заканчивается при выполнении функции 4Ch прерывания int 21h.

Вопрос: В какой точке заканчивается вызываемая программа по прерыванию Ctrl-C?

Ответ: Вызываемая программа заканчивается в месте ожидания нажатия клавиши, на функции 01h вектора прерывания 21h.

Выводы.

В ходе выполнения лабораторной работы №6 был модифицирован ранее построенный программный модуль лабораторной работы №2, а так же построен загрузочный модуль динамической структуры.