

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

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

Студент гр. 0381

Ефимов Н.Д.

Преподаватель

Губкин А.Ф.

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.

Задание.

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 какой-либо 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.

На первом шаге была спроектирована программа лабораторной работы. Для выполнения поставленной задачи были объявлены следующие переменные для вывода на экран: success – если все работает успешно, error – если такого файла нет, ctrlC_proc – если была нажата комбинация ctrl+c.

Дальше были описаны следующие процедуры:

WRITE – вывод строки на экран, Main – головная функция, exitProg – проверяет корректность выхода из дочернего модуля, freeMem – освобождает память. Функции BYTE_TO_HEX и TETR_TO_HEX переводят двоичное и четвертичные числа в шестнадцатичные.

На втором шаге получены такие результаты:

```
F:\>lab6
Locked memory address: 9FFFh
Environment address: 118Bh
There is no symbols in the command line tail
Content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Patch:
F:\LAB2.COM
aa Process was end successfully, code: 61
F:\>
```

На третьем шаге было вот это:

```
F:\>\
lab6
Locked memory address: 9FFFh
Environment address: 118Bh
There is no symbols in the command line tail
Content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Patch:
F:\LAB2.COM
♥♥ Process was end successfully, code: 03
F:\>
```

На четвертом:

```
F:\>SubCat\lab6.exe
Locked memory address: 9FFFh
Environment address: 118Bh
There is no symbols in the command line tail
Content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Patch:
F:\SUBCAT\LAB2.COM
aa Process was end successfully, code: 61
F:\>_
```

На пятом:

```
F:\>lab6
ERROR: No such file or directory
F:\LAB2.com
F:\>_
```

Выводы.

В ходе лабораторной работы был построен загрузочный модуль динамической структуры, а также модифицирован ранее построенный программный модуль. Изучены дополнительные функции работы с памятью и исследованы возможности использования интерфейса между вызывающим и вызываемым модулями по управлению и по данным.

ПРИЛОЖЕНИЕ А

ОТВЕТЫ НА ВОПРОСЫ

1) Как реализовано прерывание Ctrl-C?

При обнаружении нажатия комбинации клавиш Ctrl-C, выполняется команда int 23h. По передаваемому адресу (0000:008C), DOS распознает, что пользователь нажал Ctrl-C. Этот адрес копируется в поле PSP (Ctrl-Break Address) функциями 26H и 4Ch , а затем восстанавливается из PSP при выходе из программы.

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

В точке вызова функции 4ch int 21h.

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

Программа завершится в точке считывания символа 01h int 21h.