

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

**ОТЧЕТ**  
**по лабораторной работе №2**  
**по дисциплине «Операционные системы»**  
**Тема: Исследование интерфейсов программных модулей**

Студент гр. 0381

Ефимов Н.Д.

Преподаватель

Губкин А.Ф.

Санкт-Петербург

2022

### **Цель работы.**

Исследование интерфейса управляющей программы и загрузочных модулей. Этот интерфейс состоит в передаче запускаемой программе управляющего блока, содержащего адреса и системные данные. Так загрузчик строит префикс сегмента программы (PSP) и помещает его адрес в сегментный регистр. Исследование префикса сегмента программы (PSP) и среды, передаваемой программе.

### **Задание.**

Шаг 1. Для выполнения лабораторной работы необходимо написать и отладить программный модуль типа .COM, который выбирает и распечатывает следующую информацию:

- 1) Сегментный адрес недоступной памяти, взятый из PSP, в шестнадцатеричном виде.
- 2) Сегментный адрес среды, передаваемой программе, в шестнадцатеричном виде.
- 3) Хвост командной строки в символьном виде.
- 4) Содержимое области среды в символьном виде.
- 5) Путь загружаемого модуля.

Сохраните результаты, полученные программой, и включите их в отчет.

Шаг 2. Оформление отчета в соответствии с требованиями. В отчет включите скриншот с запуском программы и результатами.

### **Выполнение работы.**

Как и в прошлой лабораторной работе, основная часть функционала была взята из методических указаний, а именно:

TETR\_TO\_HEX - Перевод десятичной цифры в код символа

BYTE\_TO\_HEX - Перевод байта в 16-ной с/с в символьный код

WRD\_TO\_HEX - Перевод слова в 16-ной с/с в символьный код

BYTE\_TO\_DEC - Перевод байта в 16-ной с/с в символьный код в 10-ной

с/с

## WRITESTRING – Вывод строки на экран

После метки START были объявлены переменные LOCKED\_MEM\_ADRESS (адрес заблокированной памяти), ENV\_ADRESS (адрес среды), TAIL (хвост командной среды), EMPTY\_TAIL (Если хвост пуст, то выведется содержимое этой переменной (она константна)), CONTENT (Содержимое области среды), END\_STRING (просто символ конца строки), и PATH (путь загружаемого модуля).

Дальше были написаны нужные по заданию процедуры, а именно:

GET\_LOCKED\_MEMORY – обращается к элементу PSP по адресу 02h, а дальше переводит строку в 16-ричный вид, ну и после – выводит на экран ответ.

GET\_ENVIRONMENT\_ADRESS – процедура берет адрес среды из PSP по адресу 2Ch, после чего происходит описанное выше преобразование, а затем программа выводит на экран нужное значение.

GET\_COMMANDLINE\_TILE – данный фрагмент кода берет информацию о том, что располагается в хвосте командной строки. Взяв нужное, процедура проверяет, не пуст ли он. Если пуст, то происходит переход по метке ifEmpty. Если же нет, то выполняется код в метке readContentOfTail. Это цикл, который берет символы из хвоста. После того, как все символы считаны, вызывается метка printing, в которой выводится результат работы.

GET\_ENVIRONMENT\_CONTENT\_AND\_PATH – процедура, которая предназначена для вывода содержимого среды и того пути, по которому лежит исполняемый модуль. Опять находится адрес начала среды, а дальше код считывает нужные данные при помощи некоторого количества циклов и меток.

## Тестирование.

Результат работы программы:

```
F:\>lb2.com
Locked memory address: 9FFFh
Environment address: 0188h
There is no symbols in the command line tail
Content:
PATH=Z:\
COMSPEC=Z:\COMMAND.COM
BLASTER=A220 I7 D1 H5 T6
Path:
F:\LB2.COM
F:\>_
```

```
push ds
mov ax, cs
mov ds, ax
```

### **Выводы.**

В ходе лабораторной работы был исследован интерфейс управляющей программы и загрузочных модулей, а также исследован префикс сегмента программы (PSP) и среды, передаваемой программе.

## **ПРИЛОЖЕНИЕ А**

### **ОТВЕТЫ НА ВОПРОСЫ:**

#### **Сегментный адрес недоступной памяти**

##### **1. На какую область памяти указывает адрес недоступной памяти?**

На сегментный адрес основной оперативной памяти, расположенной после программы, т. е. первый байт после памяти, выделенного программе.

##### **2. Где расположен этот адрес по отношению области памяти, отведённой программе?**

Этот адрес расположен сразу после программы.

##### **3. Можно ли в эту область памяти писать?**

Да, можно, так как в DOS общее адресное пространство. Хотя обычно эту информацию используют для вычисления размера доступной памяти.

#### **Среда, передаваемая программе**

##### **1. Что такое среда?**

Среда представляет собой область памяти, в которой в виде символьных строк записаны значения переменных, называемых переменными среды (имя=значение). Здесь переменная и значение – любые текстовые величины, байт 0 завершает каждую строку. Имеется несколько стандартных переменных среды, одна из которых PATH (определяет пути к каталогам, в которых система ищет исполняемый файл). В данной задаче PATH = Z:\

## **2. Когда создается среда? Перед запуском приложения или в другое время?**

Когда одна программа запускает другую программу, то запущенная программа получает свой собственный экземпляр блока среды, который является точной копией среды родителя, но можно создать и другую среду. Следовательно изначально, среда создаётся при загрузке ОС, но перед запуском приложения, она может быть изменена в соответствии с требованиями этого приложения. То есть копируется содержимое, которое было создано при загрузке ОС и также, если это необходимо, добавляются дополнительные параметры для данной программы, поэтому прикладная программа имеет доступ к системным переменным и к переменным, включённым в данное окружение.

## **3. Откуда берется информация, записываемая в среду?**

Из системного пакетного файла AUTOEXEC.BAT, который расположен в корневом каталоге загрузочного устройства. В нём содержатся ключевых переменных окружения (наиболее известны PATH, PROMPT, COMSPEC), которые учувствуют в создании окружения в простейшем случае.