

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

**ОТЧЕТ**  
**по лабораторной работе №3**  
**по дисциплине «Операционные системы»**  
**Тема: Исследование организации управления основной памятью**

Студентка гр. 8381

Лисок М.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

### **Цель работы.**

Для исследования организации управления памятью необходимо ориентироваться на тип основной памяти, реализованный в компьютере и способ организации, принятый в ОС. В лабораторной работе рассматривается не страничная память и способ управления динамическими разделами. Для реализации управления памятью в этом случае строится список занятых и свободных участков памяти. Функции ядра, обеспечивающие управление основной памятью, просматривают и преобразуют этот список.

В лабораторной работе исследуются структуры данных и работа функций управления памятью ядра операционной системы.

### **Основные теоретические положения.**

Учет занятой и свободной памяти ведется при помощи списка блоков управления памятью MCB (Memory Control Block). MCB занимает 16 байт (параграф) и располагается всегда с адреса кратного 16 (адрес сегмента ОП) и находится в адресном пространстве непосредственно перед тем участком памяти, которым он управляет.

MCB имеет следующую структуру:

Смещение	Длина поля (байт)	Содержимое поля
00h	1	тип MCB: 5Ah, если последний в списке, 4Dh, если не последний

01h	2	Сегментный адрес PSP владельца участка памяти, либо 0000h - свободный участок, 0006h - участок принадлежит драйверу OS XMS UMB 0007h - участок является исключенной верхней памятью драйверов 0008h - участок принадлежит MS DOS FFFAh - участок занят управляющим блоком 386MAX UMB FFFDh - участок заблокирован 386MAX FFFEh - участок принадлежит 386MAX UMB
03h	2	Размер участка в параграфах
05h	3	Зарезервирован
08h	8	"SC" - если участок принадлежит MS DOS, то в нем системный код "SD" - если участок принадлежит MS DOS, то в нем системные данные

По сегментному адресу и размеру участка памяти, контролируемого этим MCB можно определить местоположение следующего MCB в списке.

Адрес первого MCB хранится во внутренней структуре MS DOS, называемой "List of Lists" (список списков). Доступ к указателю на эту структуру можно получить, используя функцию f52h "Get List of Lists" int 21h. В результате выполнения этой функции ES:BX будет указывать на список списков. Слово по адресу ES:[BX-2] и есть адрес самого первого MCB.

Размер расширенной памяти находится в ячейках 30h, 31h CMOS. CMOS это энергонезависимая память, в которой хранится информация о конфигурации

ПЭВМ. Объем памяти составляет 64 байта. Размер расширенной памяти в Кбайтах можно определить обращаясь к ячейкам CMOS следующим образом:

```
mov AL,30h ; запись адреса ячейки CMOS
out 70h,AL
in AL,71h ; чтение младшего байта
mov BL,AL ; размера расширенной памяти
mov AL,31h ; запись адреса ячейки CMOS
out 70h,AL
in AL,71h ; чтение старшего байта
; размера расширенной памяти
```

### Описание функций и структур данных.

Таблица 1 – функции управляющей программы.

Название функции	Назначение
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
WRD_TO_HEX	Переводит число AX в строку в 16 с/с, записывая получившиеся в di, начиная с младшего разряда.
PRINT	Печатает строку на экран

Таблица 2 – структуры данных управляющей программы.

Название	Тип	Назначение
MemAvl	db	Вывод строки 'Available memory: В'
ExtMem	db	Вывод строки 'Extended memory: КВ'

TableHead	db	Вывод строки 'MCB Address   MCB Type   Owner   Size   Name '			
-----------	----	--	--	--	--

### Описание работы утилиты.

Программа выводит на экран количество доступной и расширенной памяти, а также цепочку блоков управления памятью MCB. Результат работы программы представлен на рисунке 1.

```

C:\>LAB3_1.COM
Available memory: 648928 B
Extended memory: 15360 KB
MCB Address | MCB Type | Owner | Size | Name
016F        4D      0008      16
0171        4D      0000      64      DPMILOAD
0176        4D      0040     256
0187        4D      0192     144
0191        5A      0192    648912      LAB3_1
C:\>_

```

Рисунок 1 – результат работы программы lab3\_1.com.

Написанная на первом шаге программа была изменена так, чтобы программа освобождала незанимаемую ею память. В итоге был создан новый блок, отмеченный как пустой. Результат работы программы представлен на рисунке 2.

```

C:\>LAB3_2.COM
Available memory: 648928 B
Extended memory: 15360 KB
MCB Address | MCB Type | Owner | Size | Name
016F        4D      0008      16
0171        4D      0000      64      DPMILOAD
0176        4D      0040     256
0187        4D      0192     144
0191        4D      0192    12448      LAB3_2
049C        5A      0000    636448
C:\>_

```

Рисунок 2 – результат работы программы lab3\_2.com.

Программа была переписана таким образом, чтобы после освобождения памяти, программа запрашива 64 Кб. В итоге был создан еще один блок размером 64 Кб. Результат работы представлен на рисунке 3.

```
C:\>LAB3_3.COM
Available memory: 648928 B
Extended memory: 15360 KB
MCB Address | MCB Type | Owner | Size | Name
016F        | 4D       | 0008   | 16   |
0171        | 4D       | 0000   | 64   | DPMILOAD
0176        | 4D       | 0040   | 256  |
0187        | 4D       | 0192   | 144  |
0191        | 4D       | 0192   | 12560 | LAB3_3
04A3        | 4D       | 0192   | 65536 | LAB3_3
14A4        | 5A       | 0000   | 570784 | *8Й*
C:\>
```

Рисунок 3 – Результат работы программы lab3\_3.com.

Программа была переписана таким образом, чтобы программа запрашива 64 КБ, до освобождения памяти. В итоге возникает ошибка, так как уже вся память была выделена программе. Результат работы представлен на рисунке 4.

```
C:\>LAB3_4.COM
Available memory: 648928 B
Error: Allocating more memory is not possible
Extended memory: 15360 KB
MCB Address | MCB Type | Owner | Size | Name
016F        | 4D       | 0008   | 16   |
0171        | 4D       | 0000   | 64   | DPMILOAD
0176        | 4D       | 0040   | 256  |
0187        | 4D       | 0192   | 144  |
0191        | 4D       | 0192   | 13504 | LAB3_4
04DE        | 5A       | 0000   | 635392 |
C:\>
```

Рисунок 4 – Результат работы программы lab3\_4.com.

### Выводы.

В ходе выполнения лабораторной работы были исследованы структуры данных и функции управления памятью ядра операционной системы.

### **Ответы на контрольные вопросы.**

1. Что означает “доступной объем памяти”?

Ответ: это максимальный объем памяти, который может использовать программа.

2. Где МСВ блок Вашей программы в списке?

Ответ: во всех случаях программа имеет два блока управления память. По адресу 0187h находится МСВ для переменных среды, по адресу 0192h - МСВ программного блока памяти. В третьем случае появляется еще один МСВ по адресу 04A3h для управления выделенной памятью размером 64 КБ.

3. Какой размер памяти занимает программа в каждом случае?

Ответ:

Lab3\_1.com – 648928 байт.

Lab3\_2.com – 12448 байт.

Lab3\_3.com – 12560 байт.

Lab3\_4.com – 13504 байт.