**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**

|  |  |  |
| --- | --- | --- |
| Студент гр. 6381 |  | Токун М. С. |
| Преподаватель |  | Губкин А. Ф. |

Санкт-Петербург

2018

# **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 нестраничная 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.

**Необходимые сведения для составления программы.**

Учёт 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 |

Таблица 1. Структура MCB

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

# **Ход работы.**

## **Описание данных**

**AvlMem** – содержит размер доступной памяти

**ExtMem** – содержит размер расширенной памяти

**MCBHead** – содержит строку «Get List of Lists:»

**MCBType** – содержит тип MCB

**MCBSeg** – содержит сегментный адрес PSP владельца участка памяти

**MCBSize** – содержит размер блока памяти

**MCBName** – содержит окончание блока

Error1, Error2 – содержат сообщения об ошибках

## **Описание функций**

**TETR\_TO\_HEX, BYTE\_TO\_HEX, WRD\_TO\_HEX** – перевод в 16ую СС

**BYTE\_TO\_DEC, WRD\_TO\_DEC** –перевод в 10ую СС

**IDENT\_AVL\_MEM** – определение размера доступной памяти

**IDENT\_EXT\_MEM** – определение размера расширенной памяти

**IDENT\_MCB** – определение характеристики блоков управления памятью

**IDENT\_END** – определение окончания блока

**WRITE** – вывод сообщения

## **Описание алгоритма**

Программа определяет размер доступной памяти, размер расширенной памяти, а также некоторые характеристики блоков управления памятью, а затем выводит их.

## **Результат выполнения**

Шаг 1

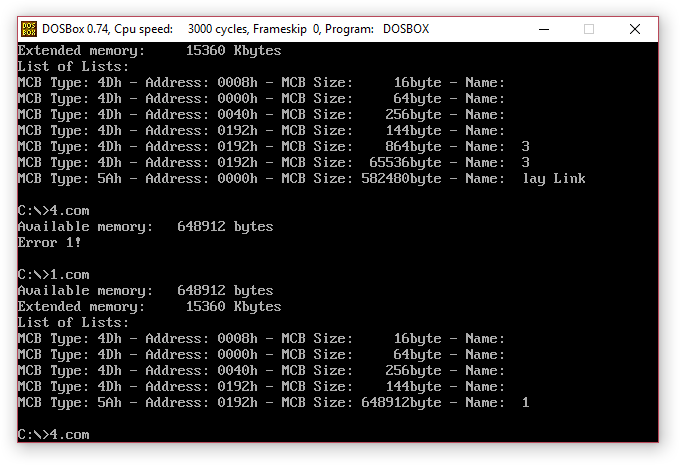


Рисунок 1 – Результат выполнения программы шага 1

Шаг 2 – добавлено освобождение памяти, которую программа не занимает

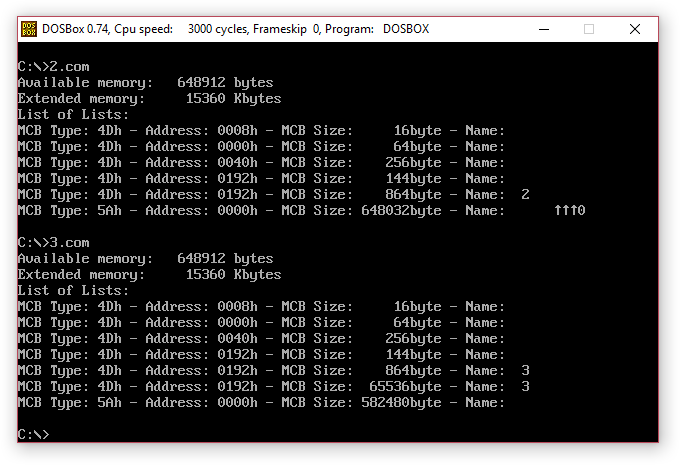


Рисунок 2 – Результат выполнения программы шага 2

Шаг 3 – после освобождения памяти программа запрашивает 64Кб памяти с помощью функции 48h прерывания 21h

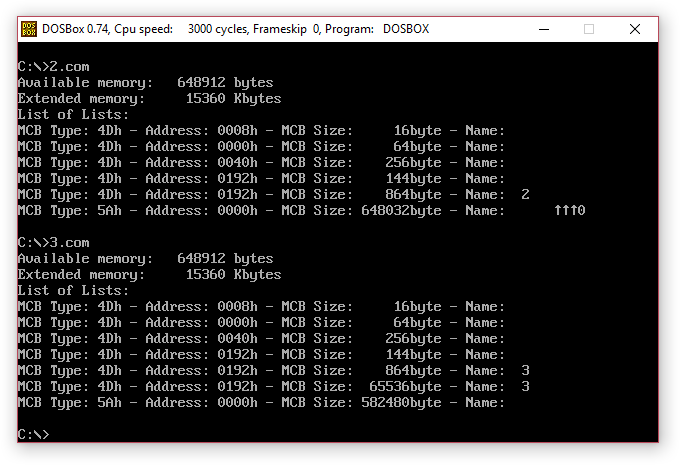


Рисунок 3 – Результат выполнения программы шага 3

Шаг 4 – запрос памяти происходит до её освобождения

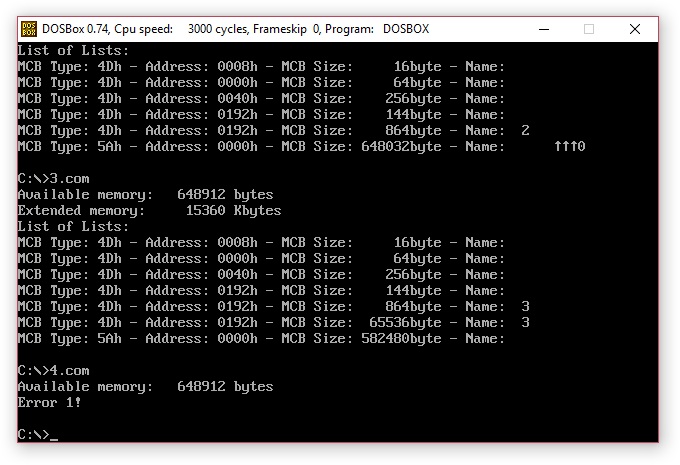


Рисунок 4 – Результат выполнения программы шага 4

Ошибка 1 сообщает о том, что ошибка, полученная путём проверки лага CF, происходит при запросе памяти

# **Вывод.**

В ходе данной лабораторной работы были исследованы структуры данных и работа функций управления памятью ядра операционной системы.

## **Ответы на контрольные вопросы.**

1. *Что означает «доступный объём памяти»?*

Доступный объем памяти означает, что это часть оперативной памяти, выделенная программе для её работы

1. *Где MCB блок Вашей программы в списке?*

MCB блок в 1-ом случае в конце списка, во 2-ом – на предпоследнем месте (после него освобождённая память), в 3-ем – третий с конца, т.к. добавляется блок запрошенной памяти

1. *Какой размер памяти занимает программа в каждом случае?*

1 – 648912 б 3 – 66400 б (т.к. 64Кб выделяются самой программой)

2 – 864 б 4 – 648912 б (начальная), далее при попытке выделить ещё - ошибка