**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МО ЭВМ**

**ОТЧЕТ**

**по лабораторной работе №3**

**по дисциплине «Операционные системы»**

**Тема: Исследование организации управления основной памятью**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 6383 |  | Лавренкова Е.Л. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

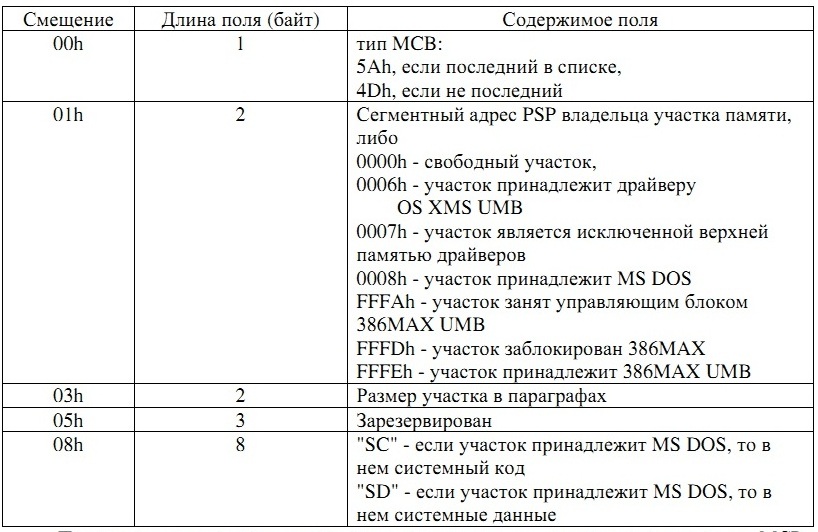
**Постановка задачи**

**Цель работы:** Исследование структур данных и работы функций управления памятью ядра операционной системы.

**Необходимые сведения для составления программы**

Учет занятой и свободной памяти ведётся при помощи списка блоков управления памятью МСВ (Memory Control Block). МСВ 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.

МСВ имеет следующую структуру (рис. 1):

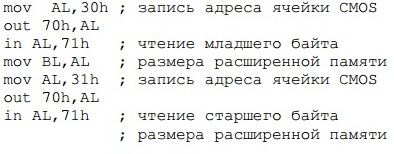


*Рис. 1*

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

Адрес первого МСВ хранится во внутренней структуре MS DOS, называемой "List of Lists" (список списков). Доступ к указателю на эту структуру можно получить используя функцию 52h "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 МСВ.

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 (рис. 2):

**

*Рис. 2*

*Порядок выполнения работы:*

* Необходимо написать и отладить программный модуль типа .COM, который выбирает и распечатывает следующую информацию:

1. Количество доступной памяти.

2. Размер расширенной памяти.

3. Выводит цепочку блоков управления памятью.

Адреса при выводе представляются шестнадцатеричными числами. Объём памяти функциями управления памятью выводится в параграфах. Необходимо преобразовать его в байты и выводить в виде десятичных чисел. Последние восемь байт МСВ выводятся как символы, не следует преобразовывать их в шестнадцатеричные числа.

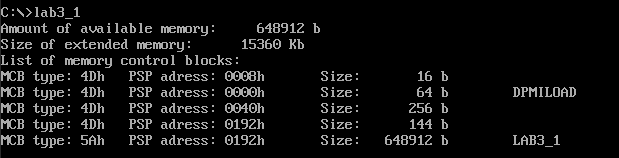
* Далее необходимо изменить программу таким образом, чтобы она освобождала память, которую она не занимает. Для этого используйте функцию 4Ah прерывания 21h (пример в разделе «Использование функции 4АН»). Хвост командной строки в символьном виде.
* Затем необходимо изменить программу еще раз таким образом, чтобы после освобождения памяти, программа запрашивала 64Кб памяти функцией 48Н прерывания 21Н.
* Далее нужно изменить первоначальный вариант программы, запросив 64Кб памяти функцией 48Н прерывания 21Н до освобождения памяти. Оформить отчёт и ответить на контрольные вопросы.

**Описание функций**

|  |  |
| --- | --- |
| **Название** | **Назначение** |
| PRINT\_STRING | Вывод строки на экран |
| TETR\_TO\_HEX | Перевод числа из 2-ой в 16-ую с/с (1/2 байта) |
| BYTE\_TO\_HEX | Перевод числа из 2-ой в 16-ую с/с (1 байт) |
| WRD\_TO\_HEX | Перевод числа из 2-ой в 16-ую с/с (2 байта) |
| BYTE\_TO\_DEC | Перевод числа из 2-ой в 10-ую с/с (1 байт) |
| WRD\_TO\_DEC | Перевод числа из 2-ой в 10-ую с/с (2 байта) |
| PRINT\_SYMB | Вывод символа на экран |

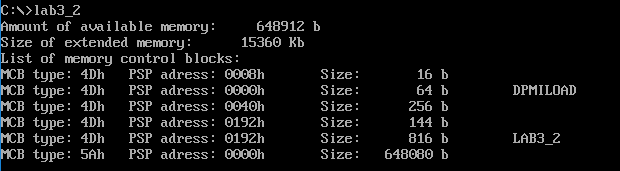
**Результат работы**

Результат работы обычной версии программы (рис. 3):



*Рис. 3*

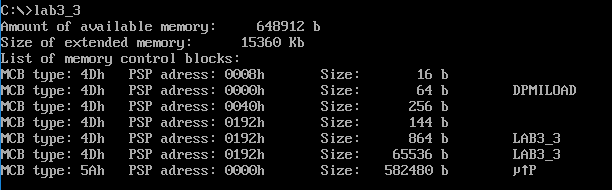
Результат работы после освобождения неиспользуемой памяти (рис. 4):



*Рис. 4*

Как можно заметить, размер блока, отведённого программе, уменьшился и образовался свободный участок памяти.

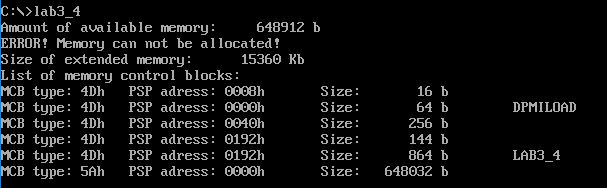
Результат работы после освобождения неиспользуемой памяти и запроса 64Кб (рис. 5):



*Рис. 5*

Как можно заметить, размер участка свободной памяти уменьшился на 64Кб и образовался участок памяти в 64Кб, запрошенной программой.

Попытка запроса выделения 64Кб памяти до освобождения неиспользуемой памяти (рис. 6):



*Рис. 6*

Попытка выделить память при отсутствии свободной памяти завершается неудачей.

**Ответы на контрольные вопросы:**

1. Что означает «доступный объем памяти»?

*Доступный объем памяти – количество памяти, не занятое другими программами, которое загрузчик выделяет программе при загрузке в основную память, после чего программа сама распоряжается этой памятью с помощью тех же функций, что использовал загрузчик. Поэтому программа может освободить не используемую память, или запросить расширение текущего объема памяти. Если программа запрашивает слишком большой объем памяти (больше, чем размер ее доступной памяти) система откажет ей в этом.*

1. Где MCB блок Вашей программы в списке?

*В первой версии программы он расположен в конце списка т.к. программа была последней загружена в память и обладает всем объемом свободной ранее памяти*

*Во второй версии, MCB блок так же седьмой по списку, но уже не последний. Последним является блок, освобожденной программой памяти*

*В третьей версии, блок так же седьмой в списке, но после него располагаются блок, памяти в 64Кб, выделенный по запросу программы и после – блок свободной памяти.*

*В четвертой версии ситуация аналогична ситуации во второй версии, так как запрос выделения памяти завершился неудачей*

1. Какой размер памяти занимает программа в каждом случае?

*В первом случае программа занимает всю выделенную ей память*

*Во втором – только объем, занимаемый самой программой*

*В третьем – объем, занимаемый самой программой и 64 Кб, выделенные ей по требованию*

*В четвертом, так же, как и во втором – только объем, занимаемый самой программой*

**Вывод**

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