

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
“ЛЭТИ” ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

Отчет
По лабораторной работе №3
По дисциплине “Операционные системы”
Тема: Исследование организации управления основной памятью

Студент гр. 8382	_____	Никитин А.Е.
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 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.

Выполнение работы.

В ходе работы был написан и отлажен программный модуль типа .com, который выбирает и распечатывает следующую информацию:

- Количество доступной памяти
- Размер расширенной памяти
- Выводит цепочку блоков управления памятью

Количество доступной памяти получено с использованием функции 4Ah с заведомо большей памятью в регистре BX, чем может быть предоставлено ОС.

Размер расширенной памяти был получен при помощи обращения к ячейкам CMOS по адресам 30h, 31h.

Для получения адреса первого MCB программа обращается к внутренней структуре MS DOS, называемой список списков. Доступ к указателю на эту структуру возможно получить используя функцию 52h. В результате выполнения этой функции EX:BS будет указывать на список списков. Слово по адресу ES:[BX-2] будет адресом самого первого MCB.

Результат выполнения программы указан на рисунке 1.

```

C:\>LAB3.COM
Available memory - 648912 b;
Extended memory - 15360 b;
Type - 4D Sector - MS DOS   Size -    16 b; Last 8 bytes -
Type - 4D Sector - FREE     Size -    64 b; Last 8 bytes - DPMILOAD
Type - 4D Sector - 0040     Size -   256 b; Last 8 bytes -
Type - 4D Sector - 0192     Size -   144 b; Last 8 bytes -
Type - 5A Sector - 0192     Size - 648912 b; Last 8 bytes - LAB3

```

Рисунок 1. Результат выполнения программы lr3-1.com

Далее программы надо было модифицировать так, что она высвобождала память, которую не использует. Сделать это можно было с помощью функции 4Ah. Результат выполнения указан на рисунке 2.

```

C:\>LAB3.COM
Available memory - 648912 b;
Extended memory - 15360 b;
Type - 4D Sector - MS DOS   Size -    16 b; Last 8 bytes -
Type - 4D Sector - FREE     Size -    64 b; Last 8 bytes - DPMILOAD
Type - 4D Sector - 0040     Size -   256 b; Last 8 bytes -
Type - 4D Sector - 0192     Size -    144 b; Last 8 bytes -
Type - 4D Sector - 0192     Size - 13712 b; Last 8 bytes - LAB3
Type - 5A Sector - FREE     Size - 635184 b; Last 8 bytes -

```

Рисунок 2. Результат выполнения программы lr3-2.com

Затем программу надо было еще раз модифицировать так, чтобы после высвобождения неиспользуемой памяти ей выделялось добавочные 64Кб памяти, с помощью функции 48h. Результат выполнения программы указан на рисунке 3.

```

C:\>LAB3.COM
Available memory - 648912 b;
Extended memory - 15360 b;
Type - 4D Sector - MS DOS   Size -    16 b; Last 8 bytes -
Type - 4D Sector - FREE     Size -    64 b; Last 8 bytes - DPMILOAD
Type - 4D Sector - 0040     Size -   256 b; Last 8 bytes -
Type - 4D Sector - 0192     Size -    144 b; Last 8 bytes -
Type - 4D Sector - 0192     Size - 13824 b; Last 8 bytes - LAB3
Type - 4D Sector - 0192     Size - 65536 b; Last 8 bytes - LAB3
Type - 5A Sector - FREE     Size - 569520 b; Last 8 bytes - XP1▲

```

Рисунок 3. Результат выполнения программы lr3-3.com

И наконец в последней модификации нужно было изменить порядок высвобождения и добавления памяти с отслеживанием регистрового флага

CF. В результате выполнения такой программы была обнаружена ошибка аллокации памяти. Результат выполнения указан на рисунке 4.

```
C:\>LAB3.COM
Available memory - 648912 b;
ERROR MEM ALLOCATION
Extended memory - 15360 b;
Type - 4D Sector - MS DOS      Size -      16 b; Last 8 bytes -
Type - 4D Sector - FREE        Size -      64 b; Last 8 bytes - DPMILOAD
Type - 4D Sector - 0040        Size -     256 b; Last 8 bytes -
Type - 4D Sector - 0192        Size -     144 b; Last 8 bytes -
Type - 4D Sector - 0192        Size -    14384 b; Last 8 bytes - LAB3
Type - 5A Sector - FREE        Size -   634512 b; Last 8 bytes -
```

Рисунок 4. Результат выполнения программы lr3-4.com

Контрольные вопросы.

1. Что означает «доступный объем памяти»?
 - Тот объем, к которому программа может обратиться для своего выполнения
2. Где MSV блок Вашей программы в списке?
 - Четвертый и пятый блоки в первой, второй и последней программах
 - В третьей программе четвертый, пятый и шестой
3. Какой размер памяти программа занимает в каждом случае?
 - Первая программа – 649056 байт.
 - Вторая программа – 13856 байт.
 - Третья программа – 79504 байта.
 - Последняя программа – 14528 байт.

Выводы.

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