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: Исследования организации управления основной памятью

Студент гр. 0381 Ефимов Н.Д.

Преподаватель Губкин А.Ф.

Санкт-Петербург
2022

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.

Задание.

Шаг 1. Для выполнения лабораторной 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. Объем памяти

функциями управления памятью выводится в параграфах. Необходимо преобразовать его в байты и выводить в виде десятичных чисел. Последние восемь байт MCB выводятся как символы, не следует преобразовывать их в шестнадцатеричные числа.

Запустите программу и внимательно оцените результаты. Сохраните результаты, полученные программой, и включите их в отчет в виде скриншота.

Шаг 2. Измените программу таким образом, чтобы она освобождала память, которую она не занимает. Для этого используйте функцию 4Ah прерывания 21h (пример в разделе «Использование функции 4AH»). Повторите эксперимент, запустив модифицированную программу. Сравните выходные данные с результатами, полученными на предыдущем шаге. Сохраните результаты, полученные программой, и включите их в отчет в виде скриншота.

Шаг 3. Измените программу еще раз таким образом, чтобы после освобождения памяти, программа запрашивала 64Кб памяти функцией 48H прерывания 21H. Повторите эксперимент, запустив модифицированную программу. Сравните выходные данные с результатами, полученными на предыдущих шагах. Сохраните результаты, полученные программой, и включите их в отчет в виде скриншота.

Шаг 4. Измените первоначальный вариант программы, запросив 64Кб памяти функцией 48H прерывания 21H до освобождения памяти. Обязательно обрабатывайте завершение функций ядра, проверяя флаг CF. Сохраните результаты, полученные программой, и включите их в отчет в виде скриншота.

Шаг 5. Оцените результаты, полученные на предыдущих шагах. Ответьте на контрольные вопросы и оформите отчет.

Выполнение работы.

Для отображения сообщений были инициализированы некоторые строки:

- ACCESSED_MEM db 13,10,'Size of accessed memory: $';
- EXTENDED_MEM db 13,10,'Size of extended memory: $';
- STRING_BYTE db ' byte $';
- STRING_MCB db 13,10,'MCB:0 $';
- ADRESS db 'Adress: $';
- ADRESS_PSP db 'PSP adress: $';
- STRING_SIZE db 'Size: $';
- MCB_SD_SC db ' SD/SC: $' ;
- STRING_ERROR db 13,10,'Memory Error!$';
- STRING_SUCCESS db 13,10,'Success!$'.

Для выполнения лабораторной работы были написаны следующие процедуры:

-TETR_TO_HEX – перевод десятичной цифры в код символа.

-BYTE_TO_HEX – перевод байта в 16-чной системе счисления в символьный код.

-WRD_TO_HEX – перевод слова в 16-чной системе счисления в символьный код

-BYTE_TO_DEC – перевод байта в 16-чной системе счисления в символьный код.

-WRITE_STRING – вывод строки на экран

-WRITE_SIZE – запись десятичного числа в строку

-PRINT_MCB - вывод информации о цепочке блоков управления памятью

-FREE_USUSED_MEMORY – освобождение неиспользуемой памяти

-GET_EXTRA_MEMORY – выделение дополнительной памяти

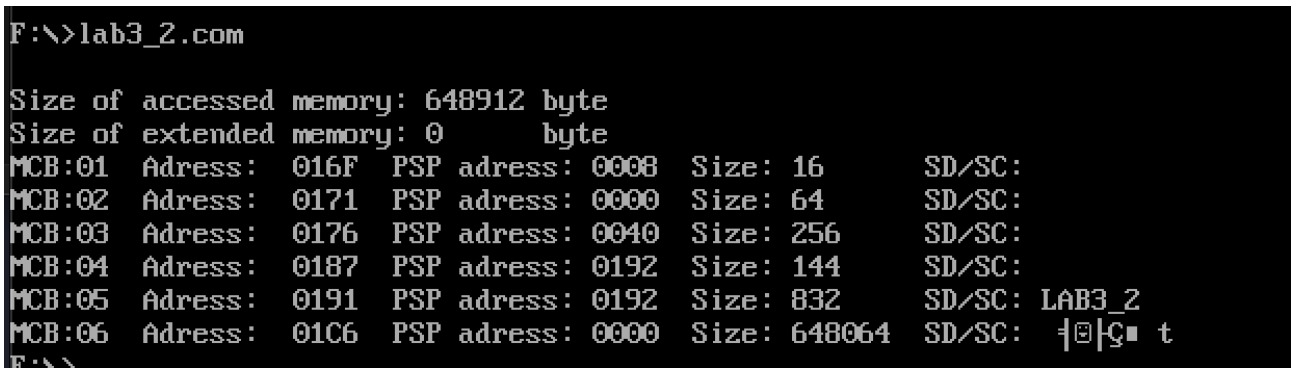
Результаты выполнения шагов изложены на скриншотах ниже:

Lab3_1:

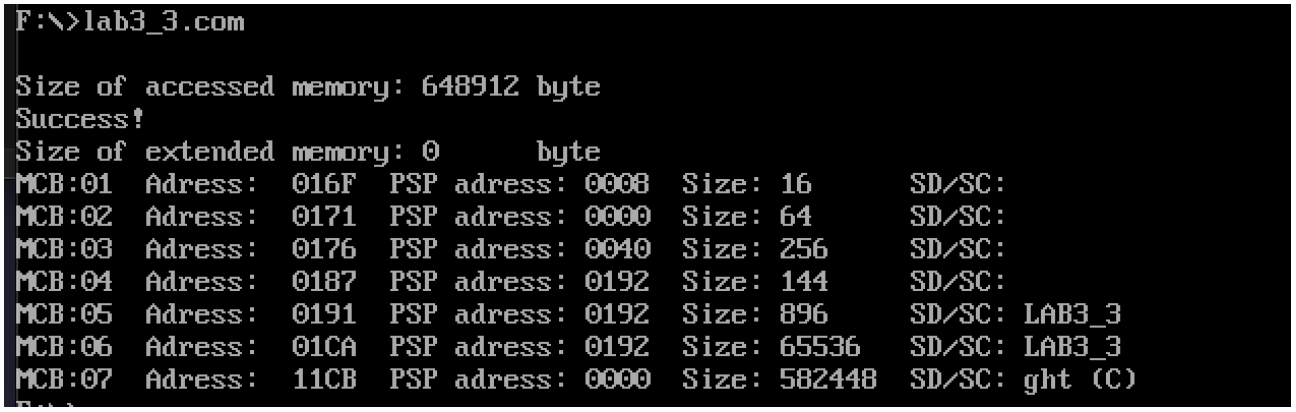
```
F:\>lab3_1.com

Size of accessed memory: 648912 byte
Size of extended memory: 0      byte
MCB:01  Adress:  016F  PSP adress: 0008  Size: 16      SD/SC:
MCB:02  Adress:  0171  PSP adress: 0000  Size: 64      SD/SC:
MCB:03  Adress:  0176  PSP adress: 0040  Size: 256     SD/SC:
MCB:04  Adress:  0187  PSP adress: 0192  Size: 144     SD/SC:
MCB:05  Adress:  0191  PSP adress: 0192  Size: 648912  SD/SC: LAB3_1
```

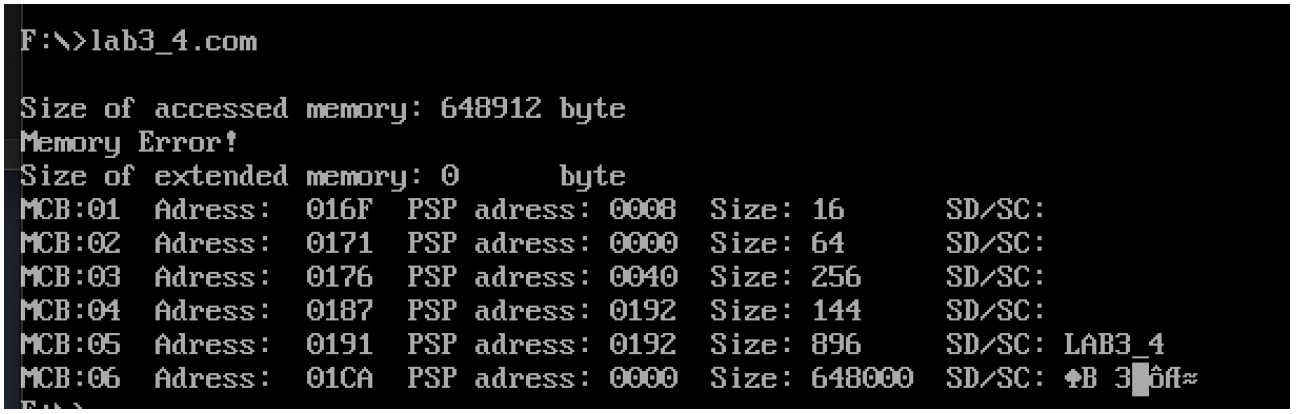
Lab3_2:



Lab3_3:



Lab3_4:



Выводы.

В ходе лабораторной работы был исследованы структуры данных и работа функций управления памятью ядра операционной системы, а затем написана утилита, которая выводит информацию об исследованной основной памяти.

ПРИЛОЖЕНИЕ А
ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что означает “Доступный объём памяти”?

Доступный объём памяти – часть оперативной памяти, выделяемой системой для корректной работы программы.

2. Где MCB блок Вашей программы в списке?

   В первом пункте задания, для файла Lab3_1, MCB находится в конце списка. Во втором – на предпоследнем, а в третьем – в пункте 5 и 6. В четвертом – в предпоследнем, как и во втором.

3. Какой размер памяти занимает программа в каждом случае?

   В первом случае – 648912.

   Во втором случае – 648064

   В третьем случае – 582448

   В четвертом случае - 648000