

**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**

Студент гр. 0381

\_\_\_\_\_

Кирильцев Д.А.

Преподаватель

\_\_\_\_\_

Ефремов М. А.

Санкт-Петербург

2022

### **Цель работы.**

Изучить организацию управления свободной памятью.

### **Постановка задачи.**

Требуется написать и отладить программные модули типа .COM, который выбирает и распечатывает следующую информацию:

1. Количество доступной памяти.
2. Размер расширенной памяти.
3. Выводит цепочку блоков управления памятью.

### **Выполнение работы.**

Таблица 1 — Процедуры в программе.

Процедура	Описание
TETR_TO_HEX	Перевод десятичной цифры в код символа
BYTE_TO_HEX	Перевод байта в 16-ной с/с в символьный код
BYTE_TO_DEC	Перевод байта в 16-ной с/с в десятичную с/с
WRD_TO_HEX	Перевод слова в 16-ной с/с в символьный код
printer	Вывод строки на экран
get_MCB	Получение информации о командном блоке памяти
get_extended_mem	Получение размера расширенной памяти
get_accessible_mem	Получение размера доступной памяти

size_calc	Вычисление размера, обработка численных данных
mem_free	Процедура освобождения памяти.
mem_alloc	Процедура выделения памяти.

Данные объявленные в программе:

AMS db 13,10,'AMS: \$'

EMS db 13,10,'EMS: \$'

units db ' byte(s)'

MCB\_number db 13,'MCB:0 \$'

HEX\_adr db 'Adress: \$'

PSP\_adr db 'PSP: \$'

size\_in\_pars db 'Paragraphs: \$'

sys\_data\_or\_code db ' System Data/Code: \$'

endl db ' ', 13, 10, '\$'

alloc\_err db 'mem\_alloc\_alloc\_error!', 13, 10, '\$'

alloc\_success db 'memory allocated successfully.', 13, 10, '\$'

У программы есть 4 версии:

1. Выводит информацию о размерах доступной и расширенной памяти, а также последовательность CMB.

```
E:\OS\LB3>lab31
EMS: 245760 byte(s)
AMS: 648912 byte(s)

MCB:0 1 Adress: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Adress: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Adress: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Adress: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Adress: 0191 PSP: 0192 Paragraphs: 648912 System Data/Code: LAB31
E:\OS\LB3>_
```

2. Выполняет вышеописанное, но также очищает использованную собой память.

```
E:\OS\LB3>lab32

AMS: 648912 byte(s)
EMS: 245760 byte(s)

MCB:0 1 Address: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Address: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Address: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Address: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Address: 0191 PSP: 0192 Paragraphs: 816 System Data/Code: LAB32
MCB:0 6 Address: 01C5 PSP: 0000 Paragraphs: 648080 System Data/Code: ♦ ïè"■-

E:\OS\LB3>_
```

3. Выполняет вышеописанное, но выделяет дополнительные 64Кб.

```
E:\OS\LB3>lab33

AMS: 648912 byte(s)
EMS: 245760 byte(s)

memory allocated successfully.

MCB:0 1 Address: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Address: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Address: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Address: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Address: 0191 PSP: 0192 Paragraphs: 928 System Data/Code: LAB33
MCB:0 6 Address: 01CC PSP: 0192 Paragraphs: 65536 System Data/Code: LAB33
MCB:0 7 Address: 11CD PSP: 0000 Paragraphs: 582416 System Data/Code:

E:\OS\LB3>
```

4. Выполняет П1, освобождает память после запроса.

```
E:\OS\LB3>lab34

AMS: 648912 byte(s)
EMS: 245760 byte(s)

mem_alloc_alloc_error!

MCB:0 1 Address: 016F PSP: 0008 Paragraphs: 16 System Data/Code:
MCB:0 2 Address: 0171 PSP: 0000 Paragraphs: 64 System Data/Code:
MCB:0 3 Address: 0176 PSP: 0040 Paragraphs: 256 System Data/Code:
MCB:0 4 Address: 0187 PSP: 0192 Paragraphs: 144 System Data/Code:
MCB:0 5 Address: 0191 PSP: 0192 Paragraphs: 928 System Data/Code: LAB34
MCB:0 6 Address: 01CC PSP: 0000 Paragraphs: 647968 System Data/Code: LAB33

E:\OS\LB3>
```

### **Ответы на вопросы**

1. Что означает “Доступный объём памяти”?

Доступный объём памяти – часть оперативной памяти выделяемой системой программе для её корректной работы.

2. Где МСВ блок Вашей программы в списке?

В первой версии МСВ - в конце списка.

Во второй - на предпоследнем месте. (Последнее место занимает высвобожденная память)

В третьей - № 5,6, т. к. мы освобождаем, а потом выделяем память.

В четвертой — предпоследний из-за ошибки выделения.

3. Какой размер памяти занимает программа в каждом случае?

В первой версии всю свободную память. (648912 байт)

Во втором только необходимый объём памяти. (816 байт)

В третьем случае необходимый объём памяти и дополнительно выделенные 64Кб объём памяти

В четвёртом случае только необходимый объём.

### **Выводы.**

Был изучен механизм организации управления свободной памятью. А также проведено знакомство с МСВ.