

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. 0381

Самойлов З. А.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

Цель работы.

Исследование организации управления основной памятью.

Постановка задачи

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 MSB 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 4АН»). 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

Таблица 1. Процедуры.

Процедура	Описание
TETR_TO_HEX	Перевод десятичной цифры в AL в код 10 с.с.
BYTE_TO_HEX	Перевод значения AL в 16 с.с.
WRD_TO_HEX	Перевод значения AX в 16 с.с.
BYTE_TO_DEC	Перевод значения AL в код 10 с.с.

Таблица 2. Макросы.

Макрос	Описание
PRINT_MSG mgs	Вывод строки через int 21h – AH = 09h
WORD_TO_BUFFER	Запись результата WRD_TO_HEX в строку данных BUFFER
HEX	Вывод «h» в консоль

Выполнение работы.

Объявлены строки для вывода информации:

- WORD_BUFFER db ' \$'
- FREE db 0DH,0AH,0DH,0AH,'Free memory: \$'
- EXTENDED db 0DH,0AH,'Extended memory: \$'
- MCB db 0DH,0AH,'Memory control block.',0DH,0AH,'\$'
- OWNER db ' Owner: \$'
- PARAGRAPHS db 0DH,0AH,' Size (paragraphs): \$'
- SD db 0DH,0AH,' SC/SD: \$'

Сначала происходит прохождение по всем доступным MCB и вывод их полей: адрес владельца, размер управляемой памяти в параграфах и код/данные с отступа 8 от начала блока. Параллельно в CX считывается количество свободной памяти блоков с адресом владельца 0000h. После прохождения по списку MCB выполняется вывод учтенной свободной памяти и размера расширенной памяти по данным из CMOS.

В результате был получен модуль:

```
DOSBox-X 0.83.20: 3000 cycles/ms, COMMAND
Main CPU Video Sound DOS Capture Drive Debug Help

Memory control block.
  Owner: 0008h
  Size (paragraphs): 0427
  SC/SD:  \_  Ёщ  \_
Memory control block.
  Owner: 072Fh
  Size (paragraphs): 0705
  SC/SD: COMMAND
Memory control block.
  Owner: 072Fh
  Size (paragraphs): 0054
  SC/SD: COMMAND
Memory control block.
  Owner: 0813h
  Size (paragraphs): 0802
  SC/SD: COMMAND
Memory control block.
  Owner: 0813h
  Size (paragraphs): 5136
  SC/SD: LB3

Free memory: 0000h
Extended memory: 3C00h
F:\>
```

Рис. 1 - .COM модуль.

При модификации из шага 2 количество МСВ увеличивается на 1.

<pre>Memory control block. Owner: 0008h Size (paragraphs): 0427 SC/SD: _ Ёщ _ Memory control block. Owner: 072Fh Size (paragraphs): 0705 SC/SD: COMMAND Memory control block. Owner: 072Fh Size (paragraphs): 0054 SC/SD: COMMAND</pre>	<pre>Memory control block. Owner: 0813h Size (paragraphs): 0802 SC/SD: COMMAND Memory control block. Owner: 0813h Size (paragraphs): 0160 SC/SD: LB3 Memory control block. Owner: 0000h Size (paragraphs): 3535 SC/SD: ength Free memory: 0000h Extended memory: 3C00h F:\></pre>
--	---

Рис. 2 - .COM модуль, шаг 2.

При модификации из шага 3 добавляется ещё один МСВ.

Memory control block. Owner: 0008h Size (paragraphs): 0427 SC/SD: _⊥ Ёщ♣ ⊥	Memory control block. Owner: 0813h Size (paragraphs): 0802 SC/SD: COMMAND
Memory control block. Owner: 072Fh Size (paragraphs): 0705 SC/SD: COMMAND	Memory control block. Owner: 0813h Size (paragraphs): 0160 SC/SD: LB3
Memory control block. Owner: 072Fh Size (paragraphs): 0054 SC/SD: COMMAND	Memory control block. Owner: 0813h Size (paragraphs): 0160 SC/SD: LB3
	Memory control block. Owner: 0000h Size (paragraphs): 1934 SC/SD:
	Free memory: 0000h Extended memory: 3C00h

Рис. 3 - .COM модуль, шаг 3.

При модификации из шага 4 добавляется 1 МСВ.

Memory control block. Owner: 0008h Size (paragraphs): 0427 SC/SD: _⊥ Ёщ♣ ⊥	Memory control block. Owner: 0813h Size (paragraphs): 0160 SC/SD: LB3
Memory control block. Owner: 072Fh Size (paragraphs): 0705 SC/SD: COMMAND	Memory control block. Owner: 0000h Size (paragraphs): 3535 SC/SD: ength
Memory control block. Owner: 072Fh Size (paragraphs): 0054 SC/SD: COMMAND	
Memory control block. Owner: 0813h Size (paragraphs): 0802 SC/SD: COMMAND	

Рис. 4 - .COM модуль, шаг 4.

Вывод.

Была исследована организация управлением основной памятью, разработана программа, которая выводит информацию о использовании памяти в системе с помощью системных функций и обращений к блокам МСВ.

Ответы на контрольные вопросы.

1) Что означает "доступный объем памяти"?

Ответ. Доступный объем памяти – размер наибольшего свободного участка памяти.

2) Где МСВ блок Вашей программы в списке?

Ответ. На шаге 1 МСВ программы является последним в списке. На 3 шаге программа занимает 2 предпоследних МСВ. На остальных шагах МСВ программы предпоследний в списке.

3) Какой размер памяти занимает программа в каждом случае?

Ответ. На шаге 1 программа занимает 5136 параграфов, т.е. 82 176 байт. На остальных шагах программа занимает 160 параграфов, т.е. 2560 байт.