МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Костромской государственный университет»

(КГУ)

ИАСТ

(наименование института)

Кафедра автоматизированных систем и технологий

(наименование кафедры)

09.03.02

Направление подготовки/Специальность Информационные системы и технологии

(наименование направления подготовки/специальности)

Дисциплина Архитектура ЭВМ

(наименование дисциплины)

Лабораторная работа №3.

Страничный механизм.

Выполнили студенты Копосов Лев Владимирович

Копосов Владимир Владимирович

(фамилия, имя, отчество)

Группа 22-ИСбо-1б

Проверил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, имя, отчество)

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_

Кострома

Вопросы:

1. Как процессор, работающий в длинном режиме, определяет соответствие между логическими адресами программы и физическими адресами в банках памяти? Опишите полный алгоритм работы с адресными таблицами.

Процессор, работающий в длинном режиме, использует механизмы управления памятью, такие как таблицы страниц, для определения соответствия между логическими адресами программы и физическими адресами в банках памяти. Блок управления памяти содержит специальный кэш, содержащий соответствие некоторых физических адресов к линейным. Если линейный адрес в нём присутствует - его физический адрес берётся из кэша, иначе обращается к каталогу таблиц, после чего физический адрес размещается в кэше. После ОС обновляет таблицу страниц и передаёт управление обратно процессору.

1. Какие состояния страниц виртуального адресного пространства вы знаете? Назовите их, и дайте краткое описание каждого состояния.

1) Свободные страницы (не используются процессом и доступны для выделения новых партий).

2) Зарезервированные таблицы (недоступны для выделения памяти, но и не отображают существующую память. К ним нельзя получить доступ, нет гарантии, что физическую память можно будет отобразить.

3) Задействованные страницы могут использоваться приложением, и ОС гарантирует, что при обращении к этим страницам они будут размещены в физ. памяти.

1. Опишите поля структуры MEMORY\_BASIC\_INFORMATION. В чем разница между полями BaseAddress и AllocationBase? Protect и AllocationProtect?

Структура MEMORY\_BASIC\_INFORMATION предоставляет информацию о регионе памяти в адресном пространстве процесса. Она содержит следующие поля:

1) BaseAddress: Это базовый адрес региона памяти. Он указывает на начало региона памяти в виртуальном адресном пространстве процесса.

2) AllocationBase: Указатель на базовый адрес диапазона страниц, выделенных функцией VirtualAlloc. Страница, на которую указывает BaseAddress, содержится в этом диапазоне выделения.

3) AllocationProtect: Это атрибуты защиты памяти, которые были установлены при выделении блока памяти для этого региона. Они могут отличаться от Protect, если атрибуты защиты были изменены после выделения блока памяти.

4) RegionSize: Размер региона, начиная с базового адреса, в котором все страницы имеют одинаковые атрибуты, в байтах.

5) State: Состояние страниц в регионе.

6) Protect: Это атрибуты защиты памяти для региона. Они определяют, как процесс может обращаться к этому региону памяти, например, может ли он читать, записывать или выполнять код в этом регионе.

7) Type: Тип страниц в регионе.

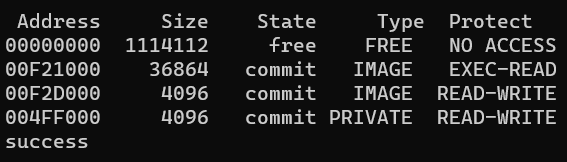
Разница между полями BaseAddress и AllocationBase заключается в том, что BaseAddress указывает на начало региона памяти в виртуальном адресном пространстве процесса, а AllocationBase указывает на начало выделенного блока памяти. Если регион памяти был выделен с помощью функции VirtualAlloc, то BaseAddress и AllocationBase будут указывать на один и тот же адрес. Однако, если регион памяти был выделен другим способом, например, загружен из файла, то BaseAddress и AllocationBase могут отличаться.

Разница между полями Protect и AllocationProtect заключается в том, что Protect указывает текущие атрибуты защиты памяти для региона, а AllocationProtect указывает атрибуты защиты, которые были установлены при выделении блока памяти. Если атрибуты защиты были изменены после выделения блока памяти, то Protect и AllocationProtect будут отличаться.

Вывод в консоль.

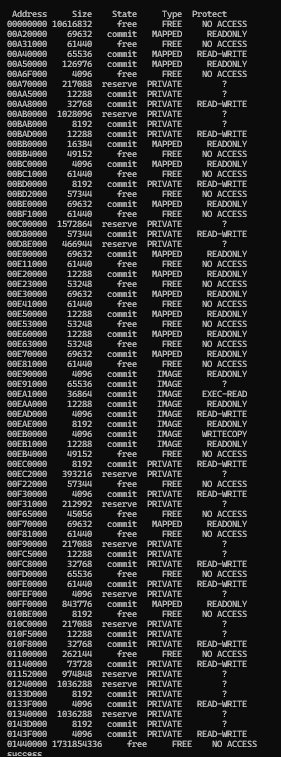
Задание 1.

Написать программу, использующую функцию VirtualQuery() для получения сведений о диапазоне страниц (структура MEMORY\_BASIC\_INFORMATION), и выводящую эти сведения на экран. Среди выводимых сведений должны присутствовать: начальный адрес, размер блока и/или конечный адрес, состояние, тип, флаги защиты. Оформите получение и вывод сведений как отдельную подпрограмму, и используйте её для получения сведений о четырёх адресах: нулевой адрес; адрес функции main() (секция кода программы); адрес переменной, описанной в main() (стек программы); адрес глобальной переменной, описанной вне main() (секция данных программы).



Задание 2.

Модифицируйте вашу программу так, чтобы она выводила на экран сведения о блоках памяти в первом гигабайте адресного пространства. Начните с нулевого адреса, а затем используйте сведения о стартовом адресе и размере блока, чтобы определить адрес следующего блока.



Дополнительное задание.

Модифицируйте программу из задания 1. Она должна выделить 1МБ памяти с помощью функции VirtualAlloc(), разрешая для него чтение и исполнение, и сохранить возвращённый функцией адрес в переменную. Сведения об этом адресе следует вывести таким же образом, как сведения об адресах из основного задания. Выведенные сведения (размер, тип защиты) должны соотвествовать тем параметрам блока, которые вы указали при вызове VirtualAlloc().

C:\Users\user\Downloads\Telegram Desktop\lab3_1b.png