

Лекция 4. Управление памятью.

Иерархия памяти.

1. Память как вид ресурса.
2. Память и ее отображение
3. Реальная память. Методы распределения памяти
 - 3.1. Распределение памяти фиксированными разделами
 - 3.2. Распределение памяти динамическими разделами
 - 3.3. Уплотнение памяти (дефрагментация памяти).

Литература

1. **Э. Таненбаум.** Современные операционные системы. 2-е изд. – СПб.: Питер, 2006. – 1038 с.:ил.
2. **Соломон Д. и Руссинович М.** Внутреннее устройство Microsoft Windows 2000. Мастер-класс / Пер. с англ. — СПб.: Питер; М.: Издательско-торговый дом «Русская Редакция», 2004. — 746 стр.: ил.
3. **Гордеев А. В.** Операционные системы: Учебник для вузов. 2-е изд. — СПб.: Питер, 2007. — 416 с.: ил.

1. Память как вид ресурса

Память – совокупность устройств, предназначенных для хранения программ, обрабатываемой информации (данных), промежуточных или окончательных результатов вычислений.

1. Память представляет собой очень важный ресурс, требующий четкого управления.
2. Размер компьютерных программ растет быстрее, чем объем памяти. Закон Паркинсона для сисадминов: **«Программы ВСЕГДА увеличиваются в размерах, заполняя всю доступную память»**.
3. В идеале каждому программисту хотелось бы иметь предоставленную только ему неограниченную по объему и скорости работы память, которая к тому же не теряет своего содержимого при отключении питания еще и совсем дешевая.

К сожалению, существующие технологии пока не могут дать нам желаемого. Может быть, способ создания такой памяти удастся изобрести именно вам.

1. Память как вид ресурса

Современная память – это концепция иерархии памяти.

Компьютеры обладают несколькими мегабайтами очень быстродействующей, дорогой и энергозависимой регистровой и кэш-памяти, несколькими гигабайтами памяти, средней по скорости, и по цене, несколькими терабайтами памяти на довольно медленных, сравнительно дешевых дисковых накопителях, и неограниченным ресурсом сменных DVD и USB. Превратить эту иерархию в абстракцию, то есть в удобную модель, а затем управлять этой абстракцией — и есть задача ОС.

Обычное время доступа

Обычный объем



1. Память как вид ресурса

Критерии классификации памяти в зависимости от назначения и особенностей реализации устройств памяти:

1. По назначению.
2. По виду физического носителя (технология производства).
3. По организации доступа (адресный: произвольный, прямой (циклический), последовательный; ассоциативный доступ).
4. По возможности записи и перезаписи.
5. По энергозависимости/энергонезависимости.
6. По типу интерфейса.
7. **По типу организации адресного пространства.**
8. По удалённости и доступности для центрального процессора (первичная, вторичная, третичная память).

1. Память как вид ресурса

Оперативная память — это важнейший разделяемый ресурс любой вычислительной системы.

Любая программа оперирует с переменными, массивами, прочими программными примитивами, т.е. обладает собственным адресным пространством.

*Логическое адресное пространство или **пространство символьных имен** состоит из имен переменных и входных точек программных модулей.*

Физическая память представляет собой упорядоченное множество ячеек реально существующей оперативной памяти. Все они пронумерованы, то есть к каждой из них можно обратиться, указав ее порядковый номер (адрес). Количество ячеек физической памяти **ограничено** и **фиксировано**.

1. Память как вид ресурса

Для размещения программы в памяти ПК необходимо **связать** каждое указанное пользователем *символьное имя* с *физической ячейкой памяти*, то есть осуществить отображение пространства имен на физическую память компьютера.

Менеджер (диспетчер) памяти - часть операционной системы, управляющая иерархией памяти. Он предназначен для эффективного выделения и распределения памяти процессам, которые в ней нуждаются, и освобождать память, когда процессы завершат свою работу.

Адресное пространство — это набор адресов, который может быть использован процессом для обращения к памяти. У каждого процесса имеется собственное адресное пространство, независимое от того адресного пространства, которое принадлежит другим процессам.

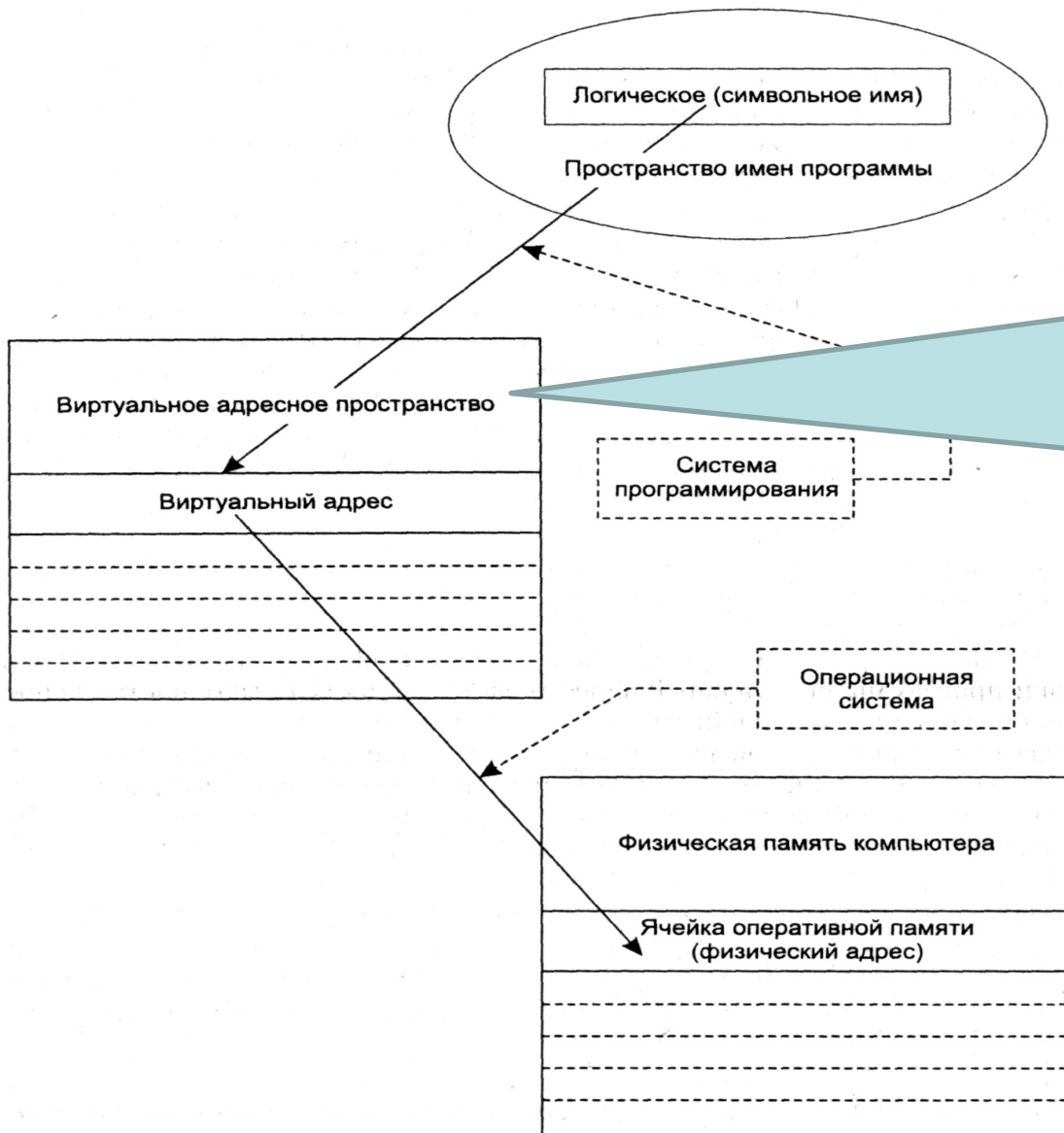
2. Память и ее отображение

Виртуальная память это технология, позволяющая выполнять процесс, который может только частично располагаться в основной памяти.

Таким образом, технология «**виртуальная память**» позволяет выполнять:

- 1) программы, размеры которых превышают размеры физического адресного пространства;
- 2) одновременно несколько программ, обладающих схожим логическим пространством имен.

2. Память и ее отображение



Виртуальное адресное пространство, или виртуальная память это множество всех допустимых значений виртуальных адресов для некоторой программы.

2. Память и ее отображение

Система управления памятью делится на 3 класса.

В зависимости от класса разной сложности бывают и менеджеры управления памятью.

- 1) Разделяющие только оперативную память между процессами
- 2) Перемещающие процессы между оперативной памятью и диском во время их выполнения, т.е. осуществляющие подкачку процессов целиком – **swapping**
- 3) Перемещающие фрагменты процессов, используя страничную подкачку – **paging**

2. Память и ее отображение

Первый способ отображения памяти.

Полная тождественность виртуального адресного пространства физической памяти ($\Lambda \neq B = \Phi$). В отображение участвует только система программирования.

Система программирования генерирует *абсолютную двоичную программу*, в этой программе все двоичные адреса таковы, что программа может исполняться только тогда, когда ее виртуальные адреса будут точно соответствовать физическим.

Пример. Несистемный загрузчик NSB из MBR - считывается в ОП, начиная с ячейки 07c0:0000h.

2. Память и ее отображение

Второй способ отображения памяти.

Тождественность виртуального адресного пространства исходному логическому пространству имен ($\Lambda = B \neq \Phi$). Отображение выполняется самой операционной системой, которая во время исполнения использует таблицу символьных имен.

Такая схема отображения используется чрезвычайно редко. Она характерна для простейших компьютерных систем, в которых вместо операционной системы использовался встроенный интерпретатор (например, Basic).

2. Память и ее отображение

Третий способ отображения памяти.

Это двухэтапное отображение адреса ($L \neq V \neq P$). В простейшем случае транслятор/компилятор генерирует *относительные* адреса, которые, по сути, являются виртуальными адресами, с последующей настройкой программы на один из непрерывных разделов. Второе отображение осуществляется перемещающим загрузчиком. После загрузки программы виртуальный адрес теряется, и доступ выполняется непосредственно к физическим ячейкам.

2. Память и ее отображение

Третий способ отображения памяти (продолжение).

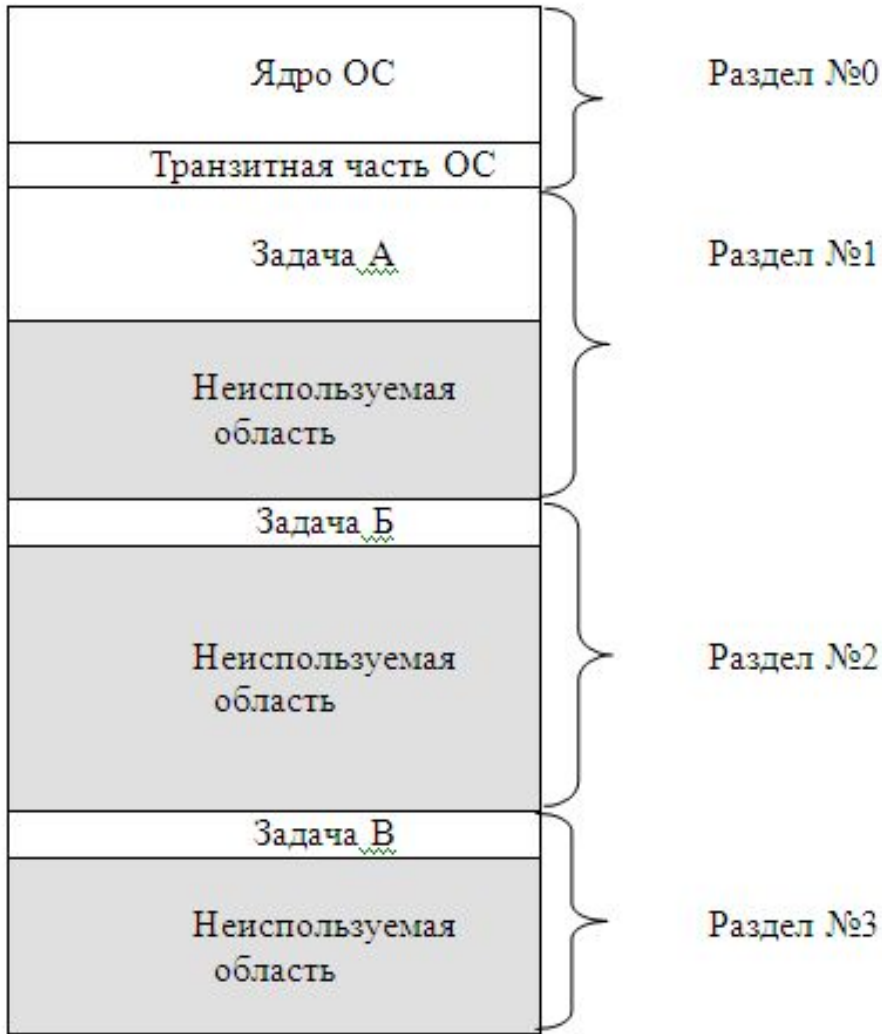
Общая схема двухэтапного отображения адресов предусматривает три варианта отображения памяти:

- объем **виртуального** адресного пространства программы V_v меньше объема **физической** памяти V_p ($V_v < V_p$), практически не встречается (мини-ЭВМ, ОС DOS, современные ПК с большой ОП);
- объем **виртуального** адресного пространства программы V_v равен объему **физической** памяти V_p ($V_v = V_p$), раньше встречалась очень часто, особенно характерна она была для недорогих вычислительных комплексов;
- объем **виртуального** адресного пространства программы V_v больше объема **физической** памяти V_p ($V_v > V_p$).

3. Реальная память. Методы распределения памяти



3.1. Распределение памяти фиксированными разделами



Разрывное распределение памяти разделами с фиксированными границам

3.1. Распределение памяти фиксированными разделами

Недостатки:

1. Неэффективное использование оперативной памяти
2. Потеря процессорного времени (в основном, из-за операций ввода-вывода).

Достоинства:

1. Простая реализация механизма защиты памяти.

3.1. Распределение памяти фиксированными разделами

Фрагментация памяти это наличие неиспользуемых областей памяти, разделенных между собой областями с загруженными программами.

Потери памяти при фиксированном ее разделении могут быть очень значительными.

Способы борьбы с фрагментацией памяти:

1. выделять раздел ровно такого объема, который нужен под текущую задачу;
2. размещать задачу не в одной непрерывной области памяти, а в нескольких областях (способы организации *виртуальной памяти*).

Разделы создаются:

- при загрузке системы;
- по мере необходимости.

3.2. Распределение памяти динамическими разделами

Выделяется раздел ровно такого объема, который нужен под текущую задачу.

Диспетчер памяти ведет учет и распределение участков. По мере удаления программ образуются свободные участки, которые сортируются либо по адресам, либо по объему.

Выделение памяти под новый раздел может осуществляться ***одним из трех основных алгоритмов***:

- первый подходящий участок;
- самый подходящий участок;
- самый неподходящий участок.

3.2. Распределение памяти динамическими разделами

Первый алгоритм – «первый подходящий участок».

Свободные участки памяти упорядочиваются по адресам. Выделяется тот, кто первым подойдет по объему (*пример*). В среднем необходимо просмотреть половину списка.

Память для небольших задач преимущественно будет выделяться в области младших адресов, и, следовательно, это увеличит вероятность того, что в области старших адресов будут образовываться фрагменты достаточно большого объема.

3.2. Распределение памяти динамическими разделами

Второй алгоритм – «самый подходящий».

Предполагается, что список свободных областей упорядочен по возрастанию объема фрагментов. Для нового раздела будет использован фрагмент свободной памяти, объем которой наиболее точно соответствует требуемому (*пример*). В среднем необходимо просмотреть половину списка.

В начале списка увеличивается количество малых разделов. При выделении нового участка количество проверяемых участков будет все более возрастать.

3.2. Распределение памяти динамическими разделами

Третий алгоритм – «самый неподходящий участок».

Список свободных областей упорядочивается по убыванию объема свободного фрагмента.

Самый неподходящий участок является самым эффективным способом распределения ОП.

3.3. Уплотнение памяти (дефрагментация памяти).

Все вычисления приостанавливаются.

Диспетчер памяти корректирует свои списки, перемещая разделы в начало памяти (или, наоборот, в область старших адресов).

Недостаток – потеря времени на уплотнение.

Контрольные вопросы

1. Что такое иерархия памяти?
2. Зачем нужен менеджер(диспетчер) памяти?
3. Что из себя представляет *физическая память*?
4. Что составляет *пространство символьных имен*?
5. Опишите процесс отображения пространства имен на физическую память компьютера.
6. Что такое виртуальная память?
7. Что такое «виртуальный адрес», «виртуальное адресное пространство»? Чем (в общем случае) определяется максимально возможный объем виртуального адресного пространства программы?
8. Опишите принципы управления памятью в однопрограммных операционных системах.
9. Что означает *оверлейные структуры программ*?

Контрольные вопросы

10. Как выполняется распределение памяти фиксированными разделами. Достоинства и недостатки.
11. Как выполняется распределение памяти динамическими разделами. Достоинства и недостатки.
12. Зачем применяется дефрагментация памяти?
13. Каковы методы распределения памяти в мультипрограммных системах?
14. Чем *статическое* распределение памяти отличается от *динамического*?
15. Что такое *свопинг*? Что такое *пейджинг*?