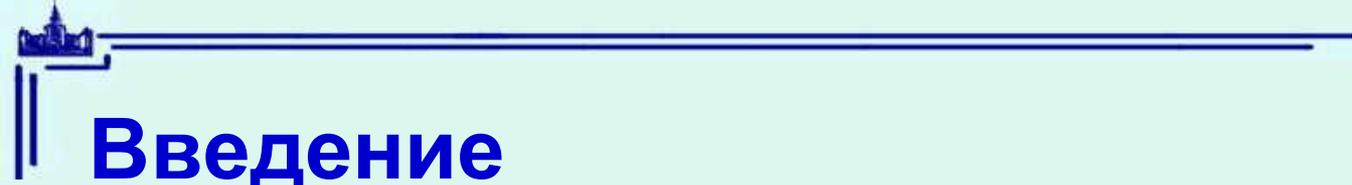


Операционные системы

лекции 7, 8

- Иерархия памяти в вычислительной системе
- Расслоение ОП
- Виртуальная память
- Способы организации виртуальной памяти



Введение

Мечта программиста:

Память ПК должна быть
неограниченной (по размеру и по скорости доступа)
энергонезависимой
дешевой

Факты:

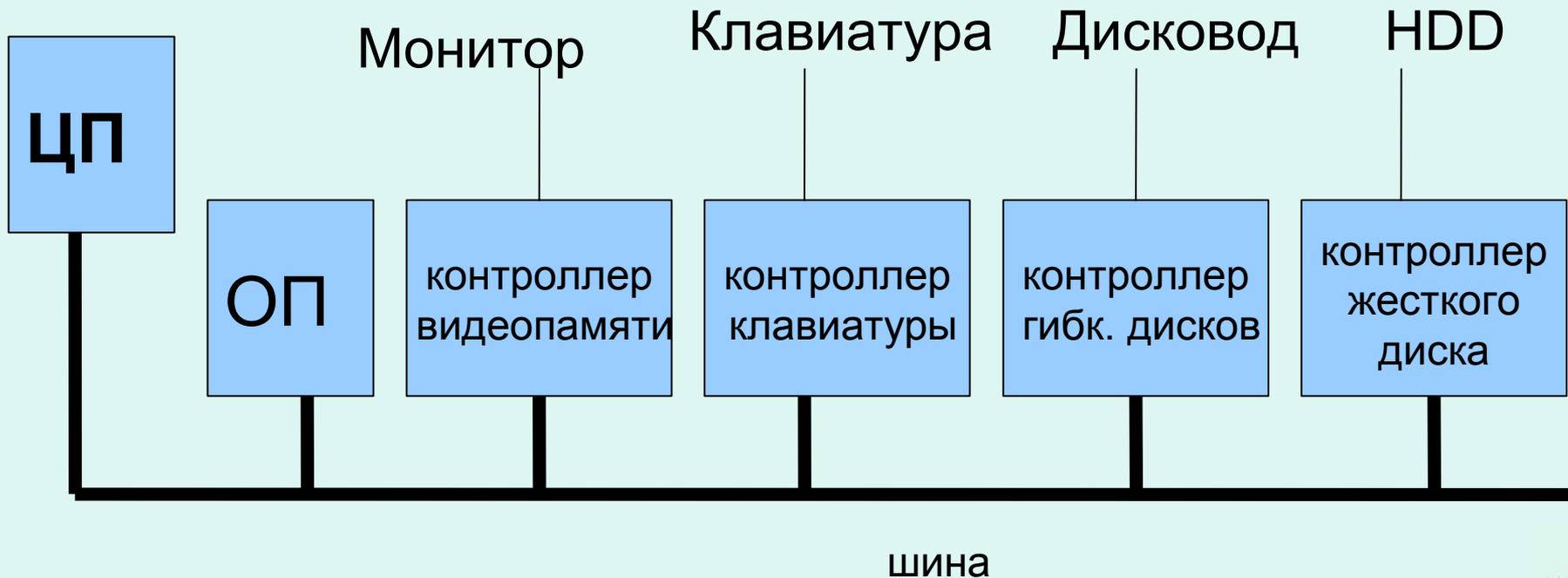
1. Память среднего домашнего компьютера 2002 г.
больше в тысячи раз
памяти IBM 7094 (самые мощные в 1960е годы)
2. Программы увеличиваются в размере быстрее, чем память

Память современного компьютера организована иерархически.



Обзор аппаратного обеспечения компьютера

Абстрактная модель ПК



Иерархия запоминающих устройств

РЕГИСТРЫ

КЭШ

ОПЕРАТИВНАЯ ПАМЯТЬ скорость

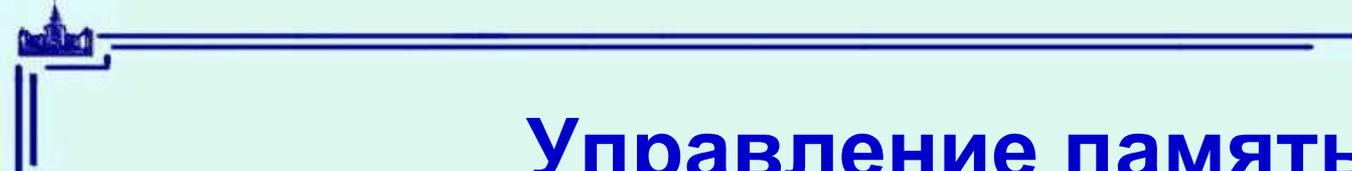
ДИСКИ

ЛЕНТЫ

объем

СТОИМОСТЬ





Управление памятью

Одна из важнейших задач ОС — распределение и управление оперативной памятью

Проблемы, решаемые менеджером памяти мультизадачной системы

1. Защита процессов друг от друга. Управление аппаратной защитой памяти
2. Недостаток объема ОП
3. Дублирование данных
4. Перемещение кода
5. Фрагментация

Swapping

Виртуальная память





Модели организации памяти: база+предел

2 регистра спец. назначения у процессора: БАЗА и ПРЕДЕЛ

В пользовательском режиме правило вычисления адресов:
 $A_{\text{физич.}} = \text{БАЗА} + A_{\text{вирт.}}$

$A_{\text{вирт.}}$ - беззнаковые, нумеруются с 0 (для любой пользовательской задачи)

$A_{\text{физич.}} < \text{ПРЕДЕЛ}$ (проверяется аппаратно)

Модификация регистров БАЗА и ПРЕДЕЛ возможна только в привилегированном режиме

Задание: оценить, какие из поставленных для менеджера памяти задач, решены





Сегментная организация памяти

Сегмент — область физической памяти, имеющая начало и длину.

Каждая задача может иметь несколько сегментов

$A_{\text{ком.}} = N + \text{смещение}$

$A_{\text{физич.}} = \text{Нач. адрес сегмента} + \text{смещение}$

Таблица сегментов

НОМЕР	АТТРИБУТЫ	НАЧ. АДРЕС	РАЗМЕР





Страничная организация памяти

Физическая память делится на кадры фиксированного размера
Виртуальная память делится на страницы того же размера

Пример

адрес — 32 разрядный

размер физической памяти = 512 Мб (2^{29} байт = $2^{17} * 2^{12}$)

2^{17} кадров по 4096 байт = 4 Кб

Сколько страниц в виртуальной памяти?

Таблица страниц

Вычисление адресов

Как выбирать размер страницы?



Страничная организация памяти: многоуровневые таблицы страниц

Пример : адрес — 32-разрядный, размер стр. 2^{12} ,

количество страниц – 2^{20} ,

каждая строка таблицы страниц - 4 байта

размер таблицы страниц каждого процесса 4 Мб

Многоуровневая таблица страниц

А вирт.

Табл.1	Табл. 2	смещение
--------	---------	----------

Табл.1 — указывает на строку, где адрес таблицы 2 уровня

Табл2. – адрес (номер строки) в таблице второго уровня

В таблице второго уровня записан номер физ. кадра,
соответствующий странице и его атрибуты

Страничная организация памяти: TLB

TLB(Translation Lookaside Buffer) — ассоциативная память, буфер быстрого преобразования адреса

действительный	вирт. стр.	изменение	защита	стр. фрейм
1	140	1	RW	31
1	20	0	RX	38
1	130	1	RW	29
1	129	1	RW	62
1	19	0	RX	50
1	21	0	RX	45
1	860	1	RW	14
1	861	1	RW	75



Страничная организация памяти: инвертированная таблица страниц

Таблица содержит по 1 записи на каждый кадр физической памяти, записано, какая вирт. страница в него отображена

Преимущество: строк меньше, т.к. меньше физических адресов.

Проблема: перевод виртуального адреса в физический.

Решение: использование TLB





Страничная организация памяти: алгоритмы замещения страниц

Замещение страницы : при возникновении страничного прерывания.

Идеи:

- 1) замена редко используемой страницы
- 2) оптимальный алгоритм.

Каждая страница помечается тем количеством команд, которое выполнится, перед первым обращением к ней. Выгружается страница с наибольшей меткой.

Проблема — значения меток не известны!

Реальные алгоритмы:

- 1) NRU (Not Recently Used)

Классы страниц:

0- не было изменений и обращений

1- не было обращений, есть изменения

2- не было изменений, есть обращения

3- произошло и обращение, и изменение

(соответствующие биты устанавливаются аппаратно, при каждом обращении к памяти, бит чтения сбрасывается по таймеру)



Страничная организация памяти: алгоритмы замещения страниц

Реальные алгоритмы:

2) FIFO (First In First Out)

В «чистом виде» : удаляется страница, которая дольше всего в памяти.

На практике (second chance): ищется самая старая страница, к которой не было обращений в предыдущем временном интервале.

0	3	7	8	12	14	15	18
---	---	---	---	----	----	----	----

A	B	C	D	E	F	G	H
---	---	---	---	---	---	---	---

к странице A зафиксировано обращение на предыд. интервале, поэтому она остается с новым значением времени

3	7	8	12	14	15	18	20
---	---	---	----	----	----	----	----

B	C	D	E	F	G	H	A
---	---	---	---	---	---	---	---

3) Часы

Формируется циклический список

Если Rбит «очередной» страницы ==0, ее заменяем, а «очередной» объявляем следующей стр.

Если Rбит==1, то Rбит=0, стр. остается, переходит к следующей.



Страничная организация памяти: алгоритмы замещения страниц

4) LRU (Least Recently Used) — вытеснение наиболее давно использовавшейся страницы

Нужен связный список всех содержащихся в памяти страниц, в начале — последние использовавшиеся стр.

Список обновляется при каждом обращении к памяти — дорогая реализация алгоритма

Дополнительная аппаратура — у каждой стр. счетчик, фиксирующий обращения, значения счетчика хранятся в таблице страниц.

При страничном прерывании ищется стр. с наименьшим значением счетчика.

Матричная реализация.

5) NFU (Not Frequently Used)

По таймеру значения Rбита добавляются к специальному счетчику — фиксируется использование страницы





Сегментно-страничная организация памяти

Задача имеет несколько сегментов

Каждый сегмент поделен на страницы

**(приписывается несколько отображений страниц в кадры
физической памяти, по одному на каждый сегмент)**

