



Лекция 2



УПРАВЛЕНИЕ ДАННЫМИ В ПАМЯТИ КОМПЬЮТЕРА



Часть 1

- Общие сведения об архитектуре памяти компьютера и специфике носителей
- 

Архитектура памяти компьютера



Оптические накопители данных

Все оптические носители информации работают по одному принципу и представляют собой пластиковые диски с отверстием в центре, процесс записи/считывания информации на/с которых осуществляется при помощи лазера.

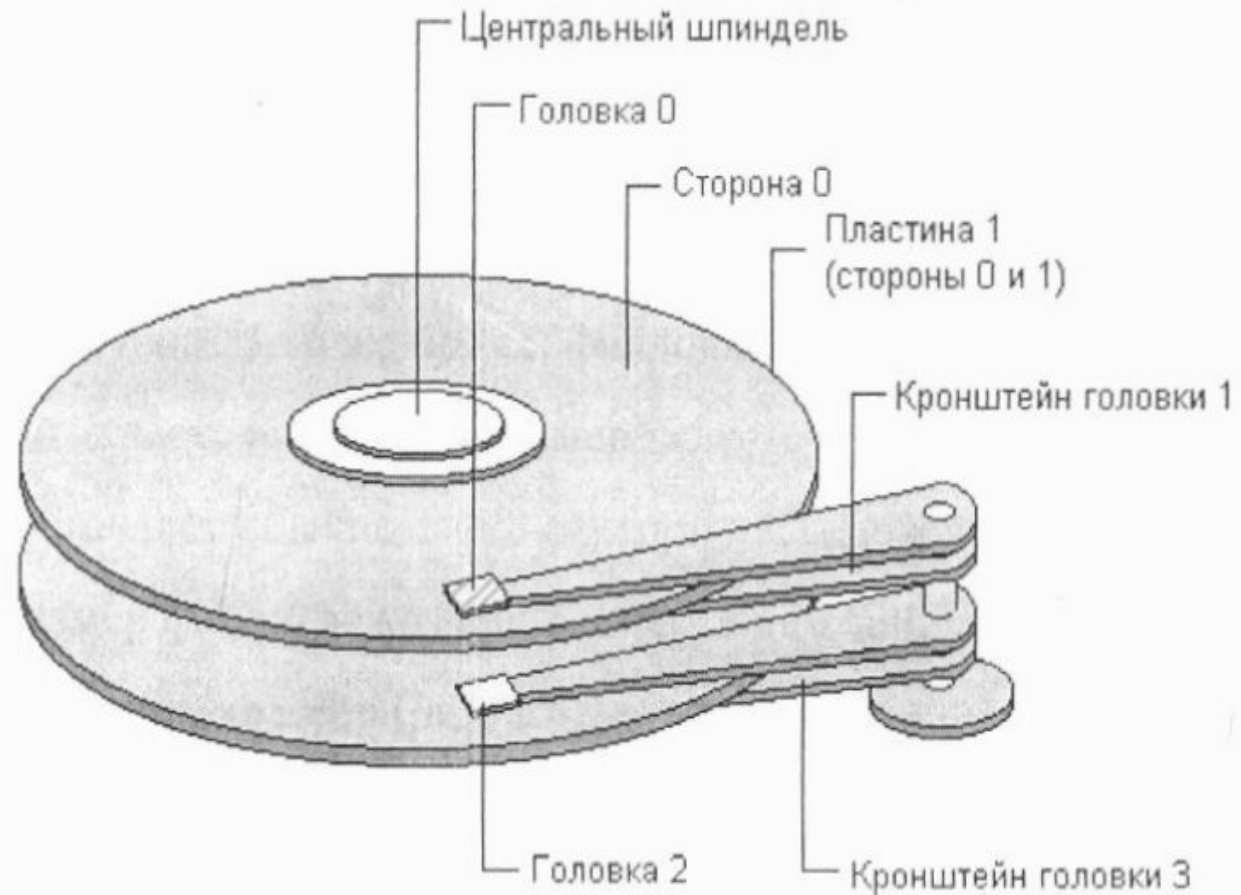
Флэш накопители

- Сегодня широкое распространение приобретает внешняя память на флэш накопителях, разработанная компанией Toshiba в 1984 году. Флеш-память хранит информацию в массиве транзисторов с плавающим затвором, называемых ячейками. На сегодняшний момент существуют две архитектурные схемы флэш-памяти: NAND и NOR

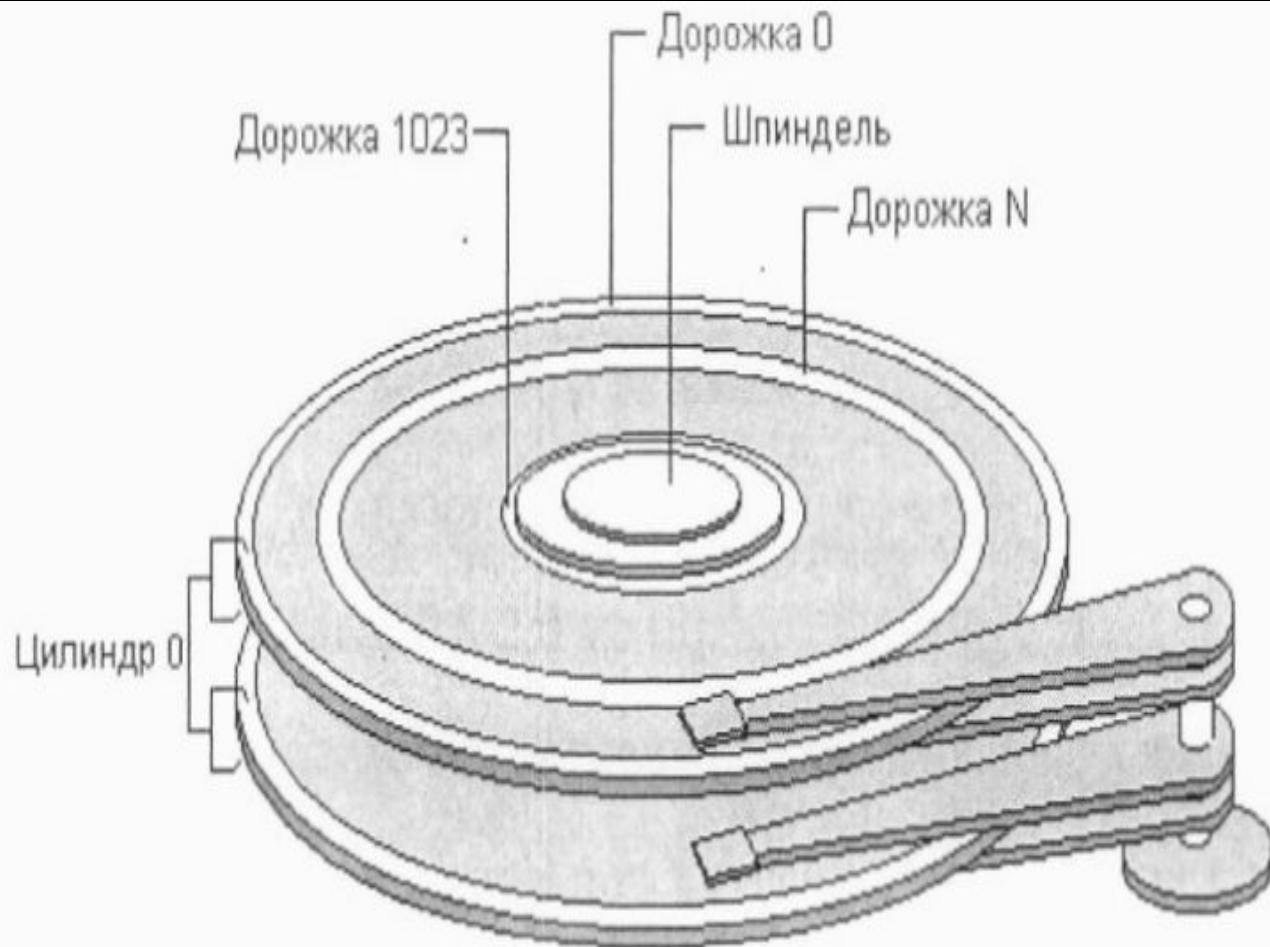
Накопители на магнитных дисках

- Наибольшим быстродействием из вышеперечисленных внешних устройств обладают накопители на магнитных дисках, однако даже современные высокопроизводительные дисковые накопители обладают миллисекундными и более задержками при обращениях к хранимым данным, за которые процессор мог бы выполнить миллионы инструкций. Разрыв производительности ленты и диска в сравнении с производительностью процессора, системной шины, шины PCI, велик и неуклонно растёт

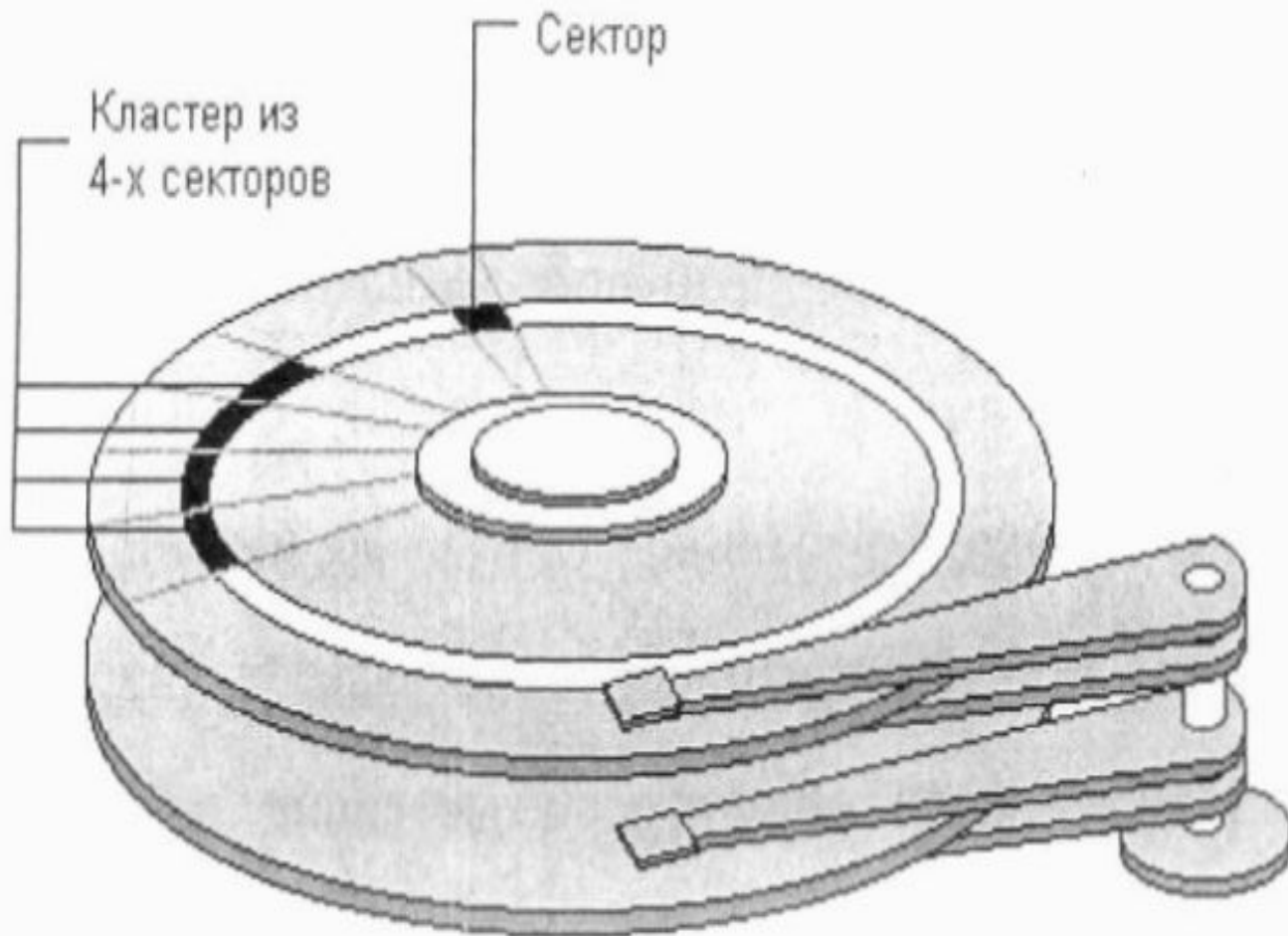
Принцип работы накопителя на магнитных дисках



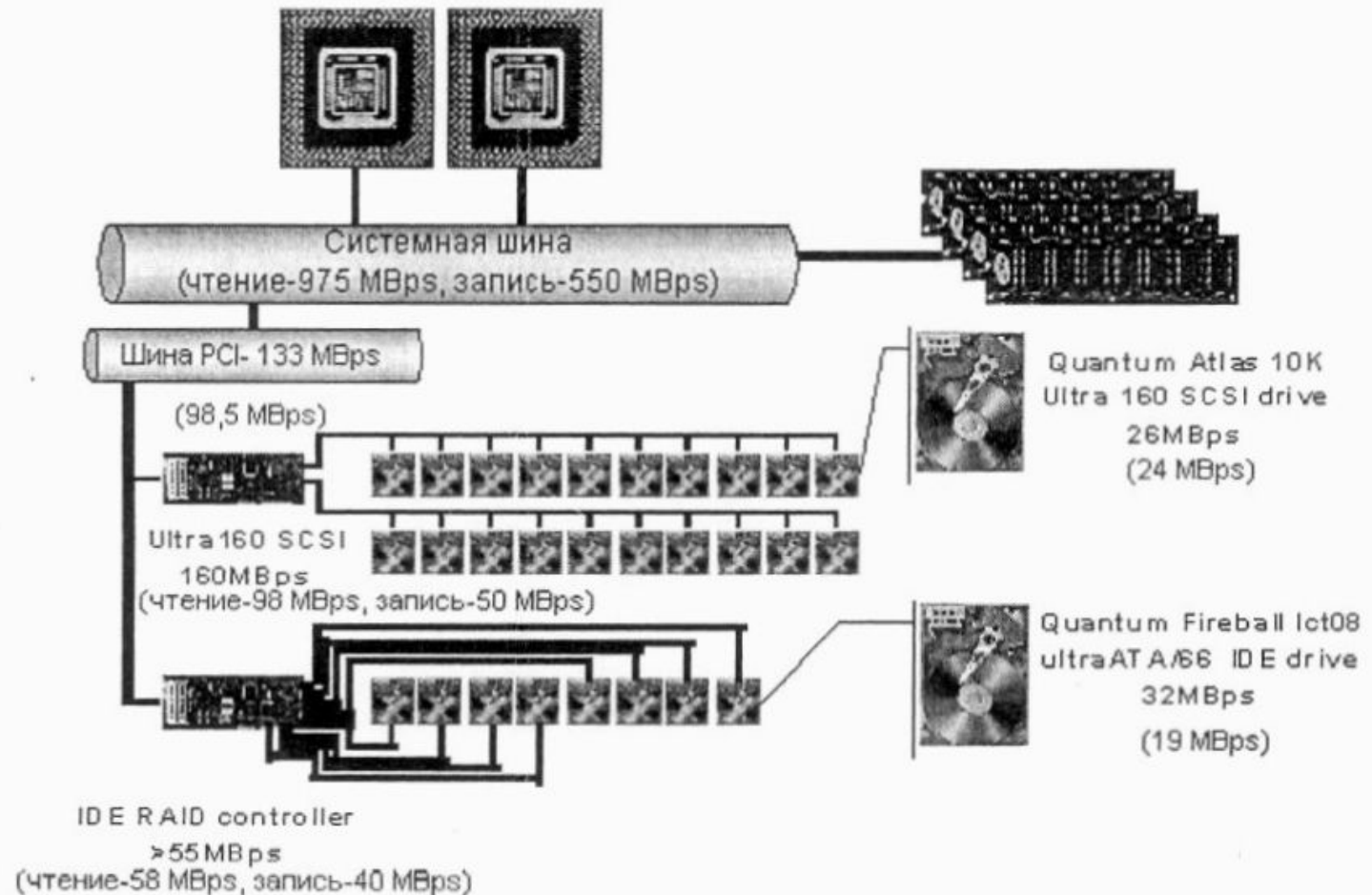
Структура дорожек и цилиндров магнитного диска



Кластеризация диска



Производительность аппаратных средств



Часть 2

- Размеры файлов и ОЗУ компьютера таковы, что в оперативной памяти можно разместить хотя бы часть файлов.

Содержательная постановка задачи

- Каждый файл, размещаемый в памяти компьютера, характеризуется двумя числами: размером и средним числом обращений к нему в единицу времени.
- Помня, что время обращения к файлу, находящемуся в ОЗУ, существенно меньше, чем время обращения к внешним носителям, требуется так разместить файлы в памяти компьютера, чтобы минимизировать среднее число обращений к внешним носителям.

Формальная постановка задачи

- Обозначения: n_i - число обращений к i -у файлу; v_i - размер i -го файла; V – размер оперативной памяти.

Задача о ранце

$$\begin{cases} \sum_i n_i z_i \rightarrow \max; \\ \sum_i v_i z_i \leq V; \\ \forall i : z_i = 1, 0. \end{cases}$$

Алгоритм решения задачи перебором

Шаг 1. $R = 0$.

Шаг 2. Выбирается ранее не просмотренная комбинация из N нулей и единиц, где N – общее число файлов. Если таковых нет – перейти к шагу 4.

Шаг 3. Если для найденной комбинации выполняется ограничение и значение S целевой функции лучше, чем R , то $R=S$, перейти к шагу 2, в противном случае – сразу переход к шагу 2.

Шаг 4. Конец алгоритма. В ячейке R – наилучшее значение целевой функции.

Пример 1

Решить перебором задачу:

$$\begin{cases} 5z_1 + 3z_2 + 7z_3 + 4z_4 \rightarrow \max; \\ 2z_1 + 4z_2 + 5z_3 + 3z_4 \leq 7; \\ \forall i : z_i = 1, 0. \end{cases}$$

Все файлы – на внешних носителях

Если время однократного обращения к внешнему носителю равно t , то суммарное время обращения ко всем носителям равно $T = 19t$.

Порядок поиска решения

№	Z_1	Z_2	Z_3	Z_4	R
1	0	0	0	1	4
2	0	0	1	0	7
3	0	0	1	1	∞
4	0	1	0	0	3
5	0	1	0	1	7
6	0	1	1	0	∞
7	0	1	1	1	∞

Далее – самостоятельно.

- Т.к. в глобально оптимальном решении на внешних носителях остаются только четные файлы, время обращения к ним равно T_1 : $T_1 = 7t$.
- Величина выигрыша во времени поиска равна η :

$$\eta = T/T_1 = 2,7.$$

Достоинства и недостатки алгоритма

- Достоинства:

1. Простота.
2. Легкость программной реализации.
3. Гарантия по завершении глобально оптимального решения.

- Недостаток: Алгоритм требует итераций .

2^N

Самостоятельно

- Решить перебором и определить величину выигрыша во времени поиска файлов:

$$\begin{cases} 5z_1 + 3z_2 + 7z_3 + 4z_4 \rightarrow \max; \\ 2z_1 + 4z_2 + 5z_3 + 3z_4 \leq 11; \\ \forall i : z_i = 1, 0. \end{cases}$$

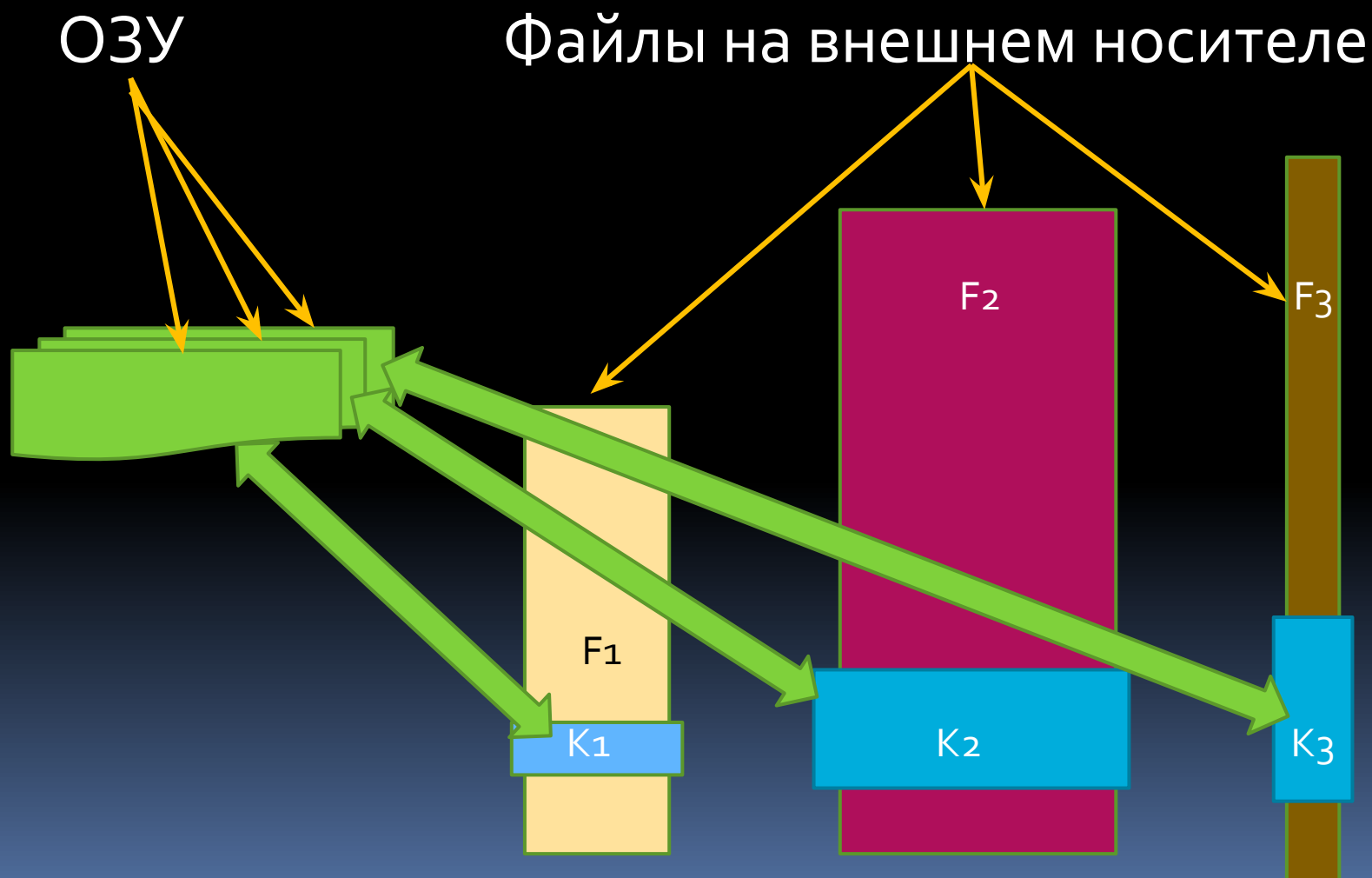
Часть 3

- Ни один из файлов не может быть размещен в оперативной памяти компьютера. Все они размещаются на внешнем носителе.

Постановка задачи кэширования данных

- Для файлов, размещенных на внешних носителях, в оперативной памяти создаются кэш-блоки, которые используются для сканирования (кэширования) «своих» файлов. Цель – минимизировать суммарное среднее число обращений к внешним носителям за планируемое время.

Иллюстрирующий пример.



Формальная постановка задачи

- Обозначения: n_i - число обращений к i -у файлу; v_i - размер i -го файла; V - размер оперативной памяти; x_i - размер i -го кэш-блока.

$$\left\{ \begin{array}{l} \sum_i \frac{n_i v_i}{x_i} \rightarrow \min; \\ \sum_i x_i \leq V; \\ \forall i : v_i \geq x_i \geq 0. \end{array} \right.$$

Пример 3 – содержательная постановка

- Даны два файла, размер первого равен 100 единиц, а второго – 200 единиц памяти. Число обращений к первому файлу в течение планового периода равно 20, ко второму файлу – 30. Объем ОЗУ равен 8,2 единиц. Дать формальную постановку и решить задачу.

Формальная постановка

- Формальная постановка

Решение

$$\left\{ \begin{array}{l} \frac{2000}{x_1} + \frac{6000}{x_2} \rightarrow \min; \\ x_1 + x_2 \leq 8,2; \\ x_1 \geq 0; x_2 \geq 0. \end{array} \right.$$

$$\left\{ \begin{array}{l} x_1 = 3; \\ x_2 = 5,2. \end{array} \right.$$