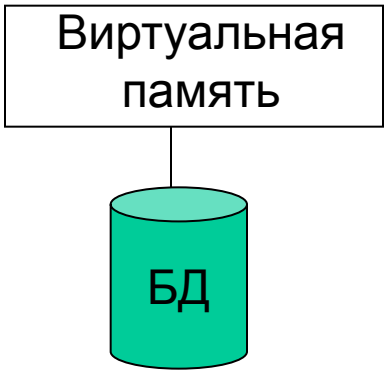
БАНКИ ДАННЫХ

Автор: Емельянов Н. Е
Правка: Тригуб Н.А.

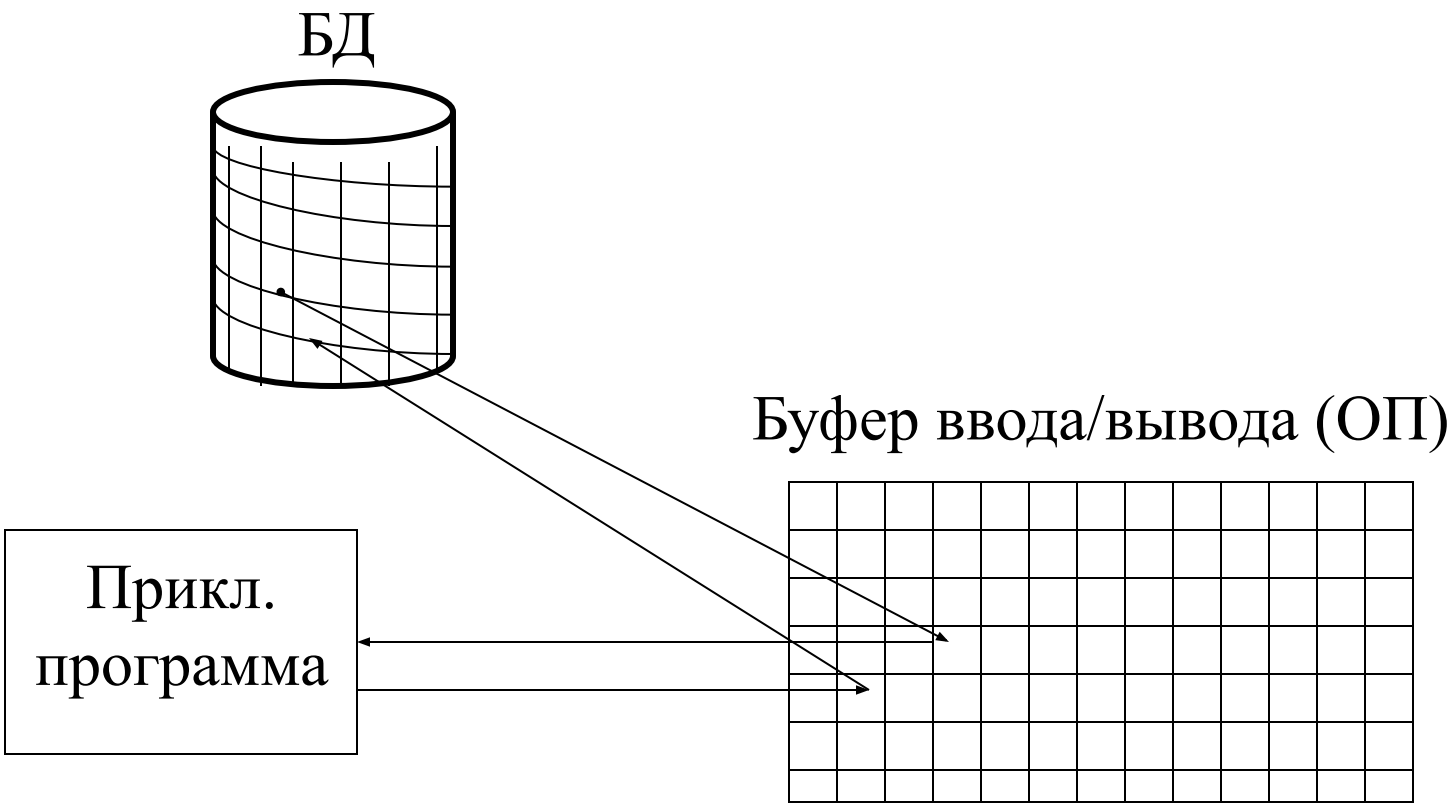
4. Организация хранения и доступа (физический уровень)

4.1. Виртуальная память



Осуществляет вызов страниц с магнитных носителей в оперативную память. Реализует механизм виртуальной памяти.

Буфер ввода/вывода



Страница БД - 2–8 Кб

Пусть БД – 100 Гб = 50 млн. страниц

Буфер ввода/вывода – 1 Гб = 500 тыс. страниц

Для эффективной работы текущий рабочий комплект страниц должен помещаться в буфер. Cash память (Cash – наличные деньги в кармане).

Справочная буфера ввода/вывода

№ стр. в / в	Идент. Стр. в БД	Время посл. обращения	Ф л а г и		
			1	2	3
1					
2					
3					

Флаги : 1 – свободна ли страница,
2 – идет ли обмен,
3 – была ли запись.

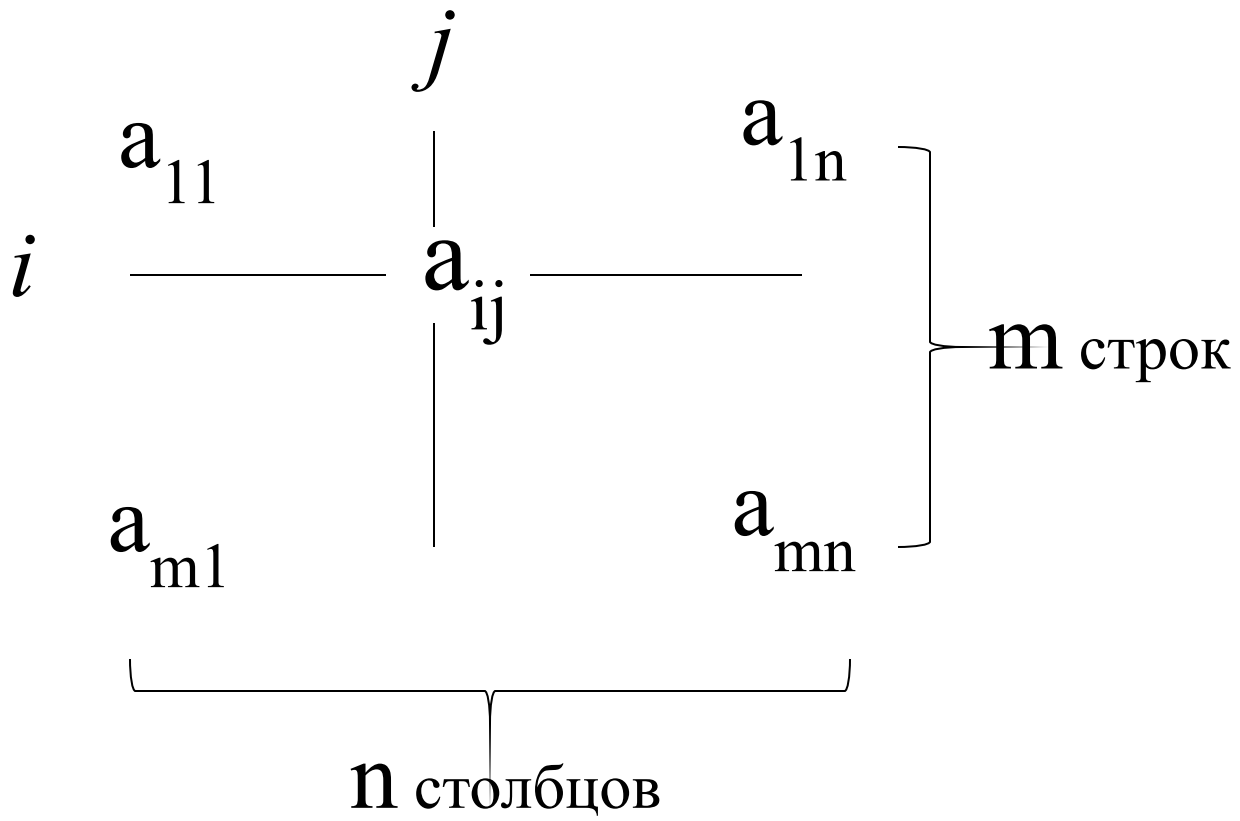
4.2. Массивы и списки

4.2.1. Однородные массивы

Массив называется однородным, если длина и формат всех элементов одинаковы

Такие массивы замечательны тем, что легко определяется адрес любого элемента

Пример двухмерного массива



Адрес a_{ij} = Адрес a_{11} + ((i –1)n + (j – 1)) d

где d – длина элементов.

4.2.2. Неоднородные массивы

Массив называется неоднородным, если длина и формат элементов могут быть разными.

Пусть элементы $a_1,\ldots,a_n$ имеют описатели $\underline{a}_1,\ldots,\underline{a}_n$

Тогда данные можно представить $(\underline{a}_1\ a_1\ ,\ldots,\ \underline{a}_n\ a_n)$

Или $(n\ \underline{a}_1\ldots\underline{a}_n\ a_1\ldots a_n)$, если все описатели $\underline{a}_i$ одинаковой длины и содержат длину данного a_i

4.2.2. dbf - формат

Если нужно описать таблицу из ***m*** строк с ***n*** атрибутами $a_1,\ldots,a_n$ в каждой строке, то в dbf -формате это будет будет представлено так

$$n\ m\ \underline{a}_1\ldots\underline{a}_{n-}(a_1\ldots a_n)_1\ \ldots\ (a_1\ldots a_n)_m$$

n m

Такой формат применил Рэтлифф в СУБД dBASE, которая быстро получила распространение, а dbf-формат стал всемирным стандартом на ~ 20 лет.

4.2.3. Языки разметки

Стандарт ISO с 1996 г.

- SGML (*Standard Generalized Markup Language*)

- XML (Extensible Markup Language)

- HTML (Hypertext Markup Language)

SGML $\supseteq$ XML $\supseteq$ HTML

- **HTML - стандарт для описания страниц в интернет**
- **В XML есть секция DTD (*Document Type Definition*), которая описывает структуру данных – аналог схемы БД**
- **В HTML заданный набор объектов, в XML можно создавать свои объекты**

Общий стиль описания:

```
<имя объекта1>
    <имя подоб1.1>
        <имя данного 1> данное 1 < /имя данного 1>
        <имя данного 2> данное 2 < /имя данного 2>
      ……
    < /имя подоб1.1>
        ……
< /имя объекта1>
```

4.3. Стеки, очереди, деки

Стек – список с включением и исключением на одном конце. Метод LIFO (Last In First Out)

Очередь – список, включение на одном конце, а исключение на другом. Метод FIFO (First In First Out)

Дек – включение и исключение на обоих концах, сокращение от Double Ended Que.

4.4. Корневые деревья

Дерево – это граф, у которого есть выделенная вершина – корень. Если его убрать, все остальные вершины распадаются на $m > 0$ поддеревьев

4.5. Графы

Граф – множество вершин, соединенных дугами (ребрами).

Графы задаются при помощи ссылок или матрицами

Матрица инцидентности

Ребра

Ребра	1	2	…	Вершины
1	1	1/0	…	1
2	1/0	1	…	2
…	…	…	…	…
n	1/0	1/0	…	n

Матрица смежности

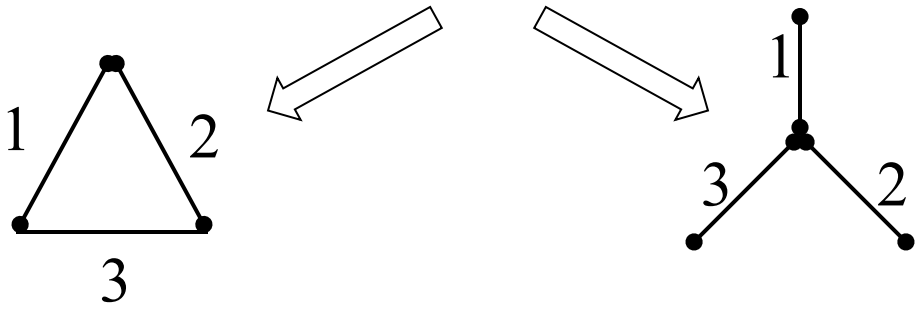
Вершины

Вершины	1	2	…
1	1/0	1	1/0
2	1/0	1/0	1
…	…	…	…
n	…	…	…

Теорема Уитни. Матрицы инцидентности и смежности однозначно определяют граф, кроме одного случая.

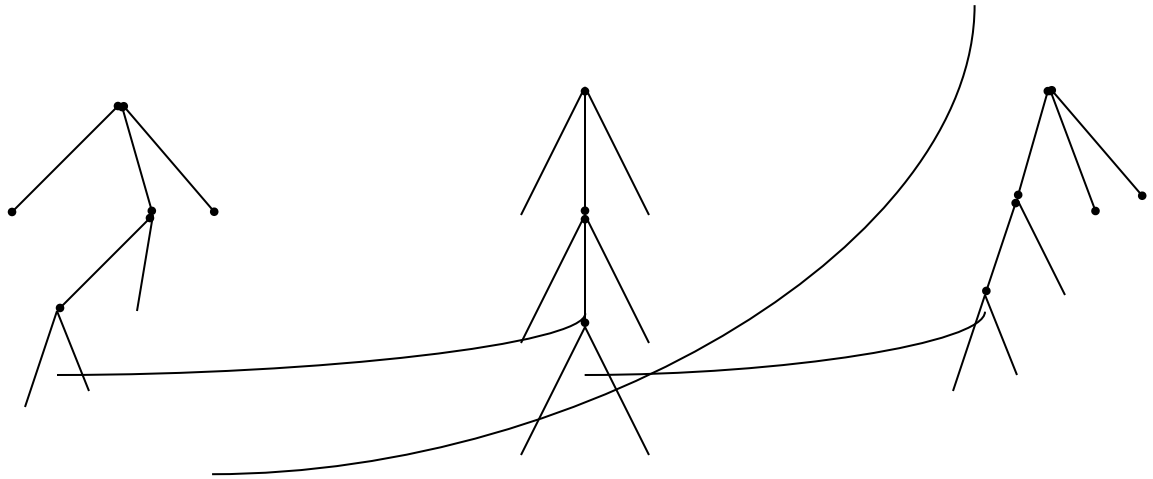
Матрица инцидентности

Ребра	1	2	3
1	1	1	1
2	1	1	1
3	1	1	1



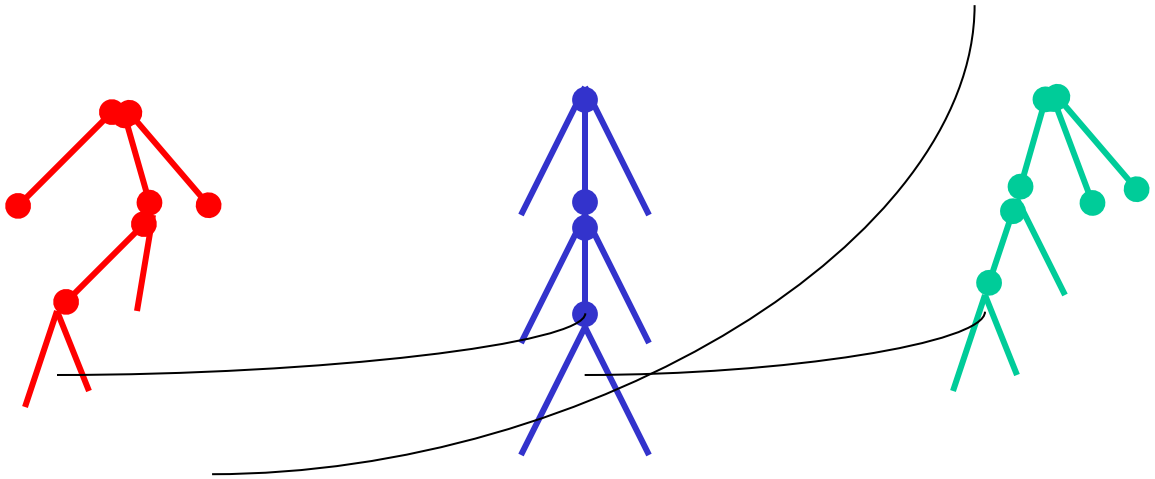
4.6. Сплетения

Сплетение – совокупность деревьев, связанных между собой ребрами.



4.6. Сплетения

Сплетение – совокупность деревьев, связанных между собой ребрами.



Раскрашенный граф – выделенных иерархий

5. Индексирование. Поиск по ключу

Индекс – единственный способ быстрого доступа.

Построение индекса – предварительная обработка информации или online поддержка при вводе данных.

5.1. Плотный индекс

Используется для неоднородных и несортированных массивов.

Массив: Идентификатор 1, данное 1; Ид 2, дан 2; Ид 3, дан 3; …. Ид n, дан n.

Плотный индекс: Ид 1, адрес 1; Ид 2, адр 2; Ид 3, адр 3; …. Ид n, адр n.

5.1. Плотный индекс

Используется для неоднородных и несортированных массивов.

Массив: Идентификатор 1, данное 1; Ид 2, дан 2; Ид 3, дан 3; …. Ид n, дан n.

Плотный индекс: Ид 1, адрес 1; Ид 2, адр 2; Ид 3, адр 3; …. Ид n, адр n.

Плотный индекс – однородный массив, который можно отсортировать и потом искать делением пополам.

5.2. Разреженный индекс

Используется для сортированных массивов.

Массив:

Ид 1^1, дан 1^1; Ид 2^1, дан 2^1; …. Ид n^1, дан n^1

Страница 1

……..

Ид 1^N, дан 1^N; Ид 2^N, дан 2^N; …. Ид n^N, дан n^N

Страница N

Индекс:

Ид 1^1, адрес стр 1; Ид 1^2, адрес стр 2; …. Ид 1^N, адрес стр N

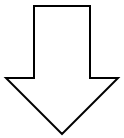
В индекс помещаются Идентификаторы первых данных всех страниц.

Область переполнения

Если при наполнении массива новое данное Ид k^i, дан k^i не помещается на страницу i, то заводится новая страница и часть страницы переносится в новую (область переполнения).

Ид 1^i, дан 1^i; …. Ид k^i, дан k^i; …. Ид n^i, дан n^i

Страница i



Ид 1^i, дан 1^i; …. Ид k^i, дан k^i;

Страница i

Ид $(k+1)^i$, дан $(k+1)^i$; …. Ид n^i, дан n^i

Страница N+m (область переполнения страницы i)

Область переполнения

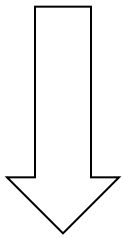
И на странице i заводится ссылка на область ее переполнения.

Ид 1^i, дан 1^i; …. Ид k^i, дан k^i; Адрес Страницы N+m

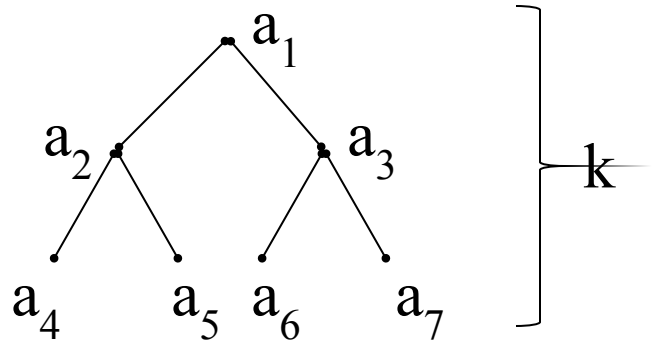
Страница i

Ид $(k+1)^i$, дан $(k+1)^i$; …. Ид n^i, дан n^i

Страница N+m (область переполнения страницы i)



5.3. B – дерево



При записи данного *x*
в B – дерево проверяем,
если *x < a1 – налево*
x >= a1 – направо,
и так во всех узлах

Обозначим
N – общее количество данных
l – количество данных на одной странице
k – глубина дерева

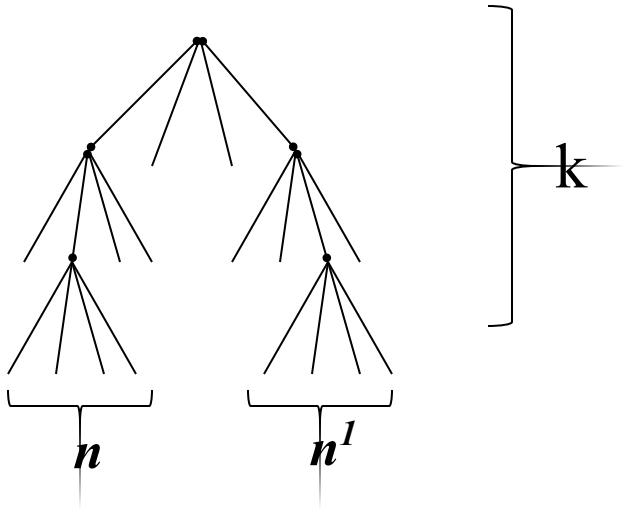
Тогда
N / l – число страниц
$2^k >= N / l$. Если ветви одной длины, то $k = \log_2 (N/l)$

Сбалансированные деревья

Определение. В – дерево называется сбалансированным по вертикали, если длины всех его ветвей отличаются не более чем на 1.

Тогда $k = [log_2 (N/l)] + 1$. Это число необходимых сравнений, в больших массивах оно обычно равно числу обменов с внешней памятью.

Определение. Дерево называется сбалансированным по горизонтали, если все веера подчиненных вершин отличаются не более чем на 1, $| n - n^1 | <= 1$.



Если ***n***-арное дерево сбалансировано по вертикали и горизонтали, то

$$k = [log_n (N/l)] + 1$$

Поэтому вместо бинарных деревьев обычно используются ***n***-арные деревья.

n - максимальное число ссылок на странице.

5.4. Хеширование

Хеш – функция

F (ключ) = идентификатор

Если из длинного ключа сделать короткий идентификатор, то возможна коллизия: F (ключ1) = F (ключ2)

На странице, определяемой идентификатором, записываются

Ключ 1, дан 1; Ключ 2, дан 2; …. Ключ n, дан n и м. б. ссылка на страницу переполнения

Сравнение методов индексирования

Метод	Число обменов	Плюсы	Минусы
Плотный индекс	Время обработки индекса + 1	Работа с несорт. и разнородными массивами	Медленно
Разреженный индекс	***N/(l*n) + 1***	Сравнительно быстро	Блоки переполнения
B – дерево	**[$log_n(N/l)$] + *1***	Быстро	Поддержка балансировки
Хеш	1 обмен, < = 3 обмена	Самый быстрый	Нет упорядоченности

N – общее число данных
l – количество данных на одной странице
n – количество ключей на одной странице