# Структура хранения данных в оперативной и внешней памяти.

## Последовательное размещение физических записей

- записи в памяти размещаются последовательно друг за другом;
- Физический адрес логической записи вычисляется по формуле:

$$(I-1)k+1$$
  
 $(I-1)k+2$   
...  
 $(I-1)k+k$   
 $I=1, 2, ..., [N/k]$ 

Где I— номер физической записи, k— номер логической записи в блоке, N— общее число логических записей

#### Поиск записи

- Поиск записи осуществляется перебором
- Читается первая физическая запись, в ОП она разбивается на *k* логических записей (разблокируется), заданное значение ключа сравнивается со значением ключа каждой логической записи. При несовпадении читается следующая физическая запись и процесс повторяется. Среднее число обращений к внешней памяти равно

$$TP = (1 + [N/k]/2]$$

## Чтение записи с заданным значением ключа

- Производится поиск
- После поиска нужная запись уже считана в оперативную память.

Среднее количество обращений к внешней памяти равно TP

#### Редактирование записи

- Производится поиск
- Логическая запись корректируется, формируется физическая запись (блок) и заносится во внешнюю память по тому адресу, откуда она была считана.
- Количество обращений к внешней памяти равно TP+1

### Удаление записи

- Производится поиск
- Логическая запись помечается как «удаленная».
- Количество обращений к внешней памяти равно TP+1

### Добавление записи

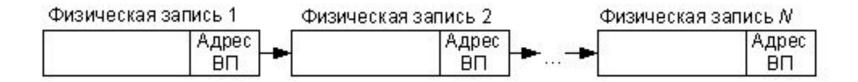
Если запись добавляется в конец таблицы, то она записывается либо в конец последней физической записи, либо сначала создается новая физическая запись. Кол-во обращений равно 1 или 2.

### Добавление записи

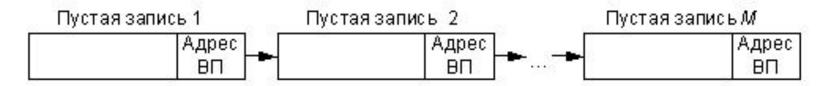
- Если запись добавляется в середину или начало таблицы, то читается физическая запись с номером [(i-1)/k]
  - Если физическая запись содержит пустые логические записи, то добавляется запись в этот блок. Количество обращений равно 2.
    - Если нет пустых записей, читается физическая запись [i/k]
      - Если в этой физической записи есть пустые логические записи, то туда добавляется логическая запись. **Кол-во обращений равно 3.**
- Иначе создается новая физическая запись, в которую добавляется логическая запись, а последующие физические записи сдвигаются.
   Среднее число обращений равно 3+[N/k]

#### Связный список

#### Связный список состоит из списка физических записей



#### и списка свободных элементов.



Каждая физическая запись содержит адрес следующей физической записи.

## Поиск записи с заданным значением ключа

Поиск записи осуществляется перебором.

Читается первая физическая запись, в ОП она разбивается на k логических записей (разблокируется), заданное значение ключа сравнивается со значением ключа каждой логической записи. При несовпадении из записи берется адрес следующей физической записи и процесс повторяется.

Среднее число обращений равно (1+[N/k])/2

## Чтение записи с заданным значением ключа

- Производится поиск
- После поиска нужная запись уже считана в оперативную память.
- Количество обращений к внешней памяти равно TP

## Чтение записи с заданным значением ключа

- Производится поиск
- После поиска нужная запись уже считана в оперативную память.
- Количество обращений к внешней памяти равно TP

#### Редактирование записи

- Производится поиск
- Логическая запись корректируется, формируется физическая запись (блок) и заносится во внешнюю память по тому адресу, откуда она была считана.
- Количество обращений к внешней памяти равно TP+1

### Удаление записи

- Производится поиск
- Логическая запись помечается как «удаленная».
- Количество обращений к внешней памяти равно TP+1

### Добавление записи

- Производится поиск записи, после которого добавляется новая запись.
- Если в этом блоке есть логическая запись, помеченная как удаленная, добавляемая запись заносится на ее место. Число обращений равно ТР+1.
- Если в этом блоке нет логических записей, помеченных как удаленные, добавляется новая физическая запись, выбираемая из списка свободных элементов.
   Соответствующие адреса переписываются.
   Число обращений равно ТР+3.

### Индексирование

- Для использования упорядочивания записей используется индексирование
- Создается индексный файл, в котором содержится индекс – таблица, состоящая из двух полей: ключевое поле и соответствующий ему адрес физической записи.
- Записи индекса упорядочены по значению ключа.

### Поиск записи с заданным значением ключа

- Из внешней памяти читается индексный файл.
- Методом половинного деления находится физическая запись.
- Эта физическая запись читается из внешней памяти.
- 4. В физической записи методом перебора находится логическая запись.

Число обращений равно 1.

## Чтение записи с заданным значением ключа

- Производится поиск
- После поиска нужная запись уже считана в оперативную память.

Число обращений равно 1.

### Редактирование записи

- Производится поиск.
- Запись корректируется и заносится на свое место.

Число обращений равно 2.

### Удаление записи

- Производится поиск.
- Запись помечается как удаленная и переписывается во внешнюю память.
- Соответствующая запись в индексе удаляется.
- 4. Измененный индексный файл переписывается во внешнюю память.

#### Число обращений равно 3.

### Добавление записи

- Добавляется новая физическая запись.
- Формируется новая запись индекса.
- Индекс переупорядочивается по значению ключа.
- 4. Измененный индекс переписывается во внешнюю память.

Число обращений равно 2.