

Задание 2

Необходимо с использованием конкретной среды разработки произвести сканирование жесткого диска и сохранить информацию о найденных файлах(каталогах), типах(так будем называть расширение), их атрибутах, размере, времени файла, а также их положение в иерархии каталогов.

По полученным данным получить статистику по следующим показателям:

1. суммарный, минимальный, максимальный, средний размер файлов
2. минимальная, максимальная дата файла
3. количество неповторяющихся типов файлов
4. количество файлов, размер которых находится в интервале верхних 10%-ах максимума размера файла (т.е. если максимум размера файла = A, то интервал от $0.9 \cdot A$ до A)

в следующих разрезах:

1. в разрезе типов файлов
2. в разрезе признака «доступный/недоступный»: отсутствие/наличие в атрибутах признаков скрытый или системный
3. в разрезе градаций размеров:
 - a. 1 группа: $0 \leq \text{size} < 1\text{кБ}$
 - b. 2 группа: $1\text{кБ} \leq \text{size} < 10\text{кБ}$
 - c. 3 группа: $10\text{кБ} \leq \text{size} < 50\text{кБ}$
 - d. 4 группа: $50\text{кБ} \leq \text{size} < 1\text{МБ}$
 - e. 5 группа: $1\text{МБ} \leq \text{size}$
4. В разрезе «фрагментов» даты (год 2007, 2006 и т.д; январь 2007, февраль 2005 и т.д.; последняя неделя и т.д.).
5. В разрезе каталогов, с учетом подкаталогов и без.

Наиболее интересна реализация разрезов 3, 4, и, особенно, 5.

Интерфейс представления результата неважен, достаточно для каждого из разрезов отобразить список элементов разреза с соответствующими показателями.

Дополнительно можно реализовать поиск файлов-дубликатов по следующим критериям: без учета имени и даты файла, учитывать только размер, тип, и, только для небольших файлов, идентичность содержимого.

Замечание: при обработке необходимо обязательно использовать таблицы, курсоры, индексы; также при обработке таблиц/курсоров по возможности меньше использовать команды и больше SQL-запросы. НЕ обязательно реализовывать все – главное способы реализации и, что немаловажно, оптимальность алгоритма.