

Лабораторная работа №2

1. Задание:

Создать таблицу, содержащую не менее 40-ка записей (тип – запись с вариантами). Упорядочить данные в ней по возрастанию ключей, где ключ – любое невариантное поле (по выбору программиста), используя: а) саму таблицу, б) массив ключей (возможность добавления и удаления записей в ручном режиме обязательна). Ввести список литературы, содержащий фамилию автора, название книги, издательство, количество страниц, вид литературы (1: техническая – отрасль, отечественная, переводная, год издания; 2: художественная – роман, пьеса, стихи). Вывести список всей переводной лит-ры по указанной отрасли с годом издания не позднее указанного.

2. ТЗ:

Цель работы – приобрести навыки работы с типом данных «запись» (структура), содержащим вариантную часть (тип литературы), а также работы с данными типа ключ-значение.

Входные данные.

На входе программа получает число записей в таблице и саму таблицу (вводится либо с консоли либо из файла) - общие данные:

- название книги
- автор
- название издательства
- количество страниц
- тип литературы.

В зависимости от типа литературы (техническая или художественная), вводятся соответствующие данные:

- год издания (для технической)
- отрасль (для технической)
- язык книги (для технической)
- область литературы (для художественной)

Также программа получает на вход номер операции, проводимой над таблицей (add / delete / filter / sort (key) / sort(table) / print a book? (1/2/3/4/5/6/quit=0)).

Выходные данные.

На выходе программа выводит таблицу с данными, по умолчанию отсортированную по порядку (index). По запросу пользователя, программа выводит отсортированную таблицу (тип сортировки выбирает пользователь. Возможно 4 типа сортировки - 2 на сортировку по всей таблице и 2 на сортировку по таблице ключей. При добавлении или удалении данных из таблицы программа выводит итоговую таблицу (с применением выбранных операций). Сортировка проводится по количеству страниц в книге. Также, по выбору пользователя,

возможна фильтрация таблицы - будут выведены только те данные, которые подходят под введенные пользователем фильтры. После сортировки таблицы программа выводит время, которое было потрачено на сортировку.

Обработка ошибок.

В случае отсутствия исходного файла с данными программа сообщит об ошибке пользователю. Если в поле введены данные размера, превышающего допустимый программой размер (20 символов), в таблице будут выведены первые 20 символов.

3. СД:

В программе была использована специально созданная структура book, содержащая информацию о книге (ее параметры). Год издания и количество страниц должны быть целыми числами. Все прочие параметры не должны превышать 15 символов (ограничение было сделано для того, чтобы было возможно вывести таблицу красиво, без переносов). Кроме общих данных, для каждого типа литературы (техническая или художественная), вводятся свои данные.

Структура book для книг:

- int id; -id книги
- char name[20]; -имя книги
- char author[15]; -автор книги
- char ph[15]; -название издательства
- int page_quantity; -количество страниц в книге
- char type[15]; -тип книги
- char language[15]; -язык книги (для технической)
- char branch[15]; -отрасль книги (для технической)
- int yop; -год публикации (для технической)
- char kind[15]; -область книги (для художественной).

4. Алгоритм:

Было реализовано 2 типа сортировки - сортировка пузырьком и выбором. Сортировка проводилась как по всей таблице, так и по таблице ключей. При удалении и добавлении записей старый массив структур сохраняется, после чего освобождается его память (n структур). Далее, данному массиву резервируется память (n+1 или n-1) в зависимости от операции (удаление, добавление) и в него копируются старые элементы.

5. Тесты:

Входные данные:	Выходные данные:	Что проверяет:
war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel фильтры: 1869, translated		Фильтрация по введенным ключам

Parusniki; Simon; 2015; AST; 736; Technical; Technical; Translated фильтры: 1700, Technical; Translated	Parusniki; Simon; 2015; AST; 736; Technical; Technical; Translated	Фильтрация по введенным ключам
Parusniki; Simon; 2015; AST; 736; Technical; Technical; Translated фильтры: 2017, Technical; Translated		Фильтрация по введенным ключам
war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel DELETE 1	war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel	Удаление
war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel ADD Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel	war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel	Добавление
1984; Orwell; AST; 352; Artistic; Novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel	war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel 1984; Orwell; AST; 352; Artistic; Novel Time of sort by keys (пузырьком) = 238 Time of sort by keys (выбором) = 309	Сортировка по таблице ключей
1984; Orwell; AST; 352; Artistic; Novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel	war and peace; Tolstoy; Russkiy vestnik; 1789; artistic; novel Idiot; Dostoyevskiy; Vestnik; 1200; artistic; novel 1984; Orwell; AST; 352; Artistic; Novel Time of sort by table (пузырьком) = 592 Time of sort by table (выбором) = 632	Сортировка по таблице

6. Выводы:

В результате работы была реализована функция сортировки элементов таблицы. Также была освоена работа со структурами, их хранением в массиве, удалением, добавлением. Для сортировки гораздо выгоднее использовать таблицу ключей, так как на ее обработку уходит в 2 раза меньше времени. Правда, программе потребуется больше памяти - придется создать таблицу ключей = $n * (\text{int} + \text{нужное поле})$

7. Вопросы:

1) Как выделяется память под вариантную часть записи?

- Выделяется область памяти, равная размеру максимального поля вариантных частей.

2) Что будет, если в вариантную часть ввести данные, несоответствующие описанным?

- Ответственность за правильный ввод данных несет программист. Данные не проверяются.

3) Кто должен следить за правильностью выполнения операций с вариантной частью записи?

- Программист.

4) Что представляет собой таблица ключей, зачем она нужна?

- Таблица ключей содержит индекс элемента и значение некоторого поля. Это позволяет ускорить поиск и сортировку исходной таблицы.

5) В каких случаях эффективнее обрабатывать данные в самой таблице, а когда – использовать таблицу ключей?

- Если в таблице много записей (или записи большого размера), удобно использовать таблицу ключей.

6) Какие способы сортировки предпочтительнее для обработки таблиц и почему?

- При сортировке предпочтительно использовать таблицу ключей.