

Лабораторная работа №1

Тема: «Классы языка C++. Разработка простейшей базы данных»

Разработать диалоговую программу для работы с простейшей базой данных в соответствии с индивидуальным заданием. Программа должна выводить меню, пункты которого описывают основных функции для работы с БД:

- Загрузка БД из файла
- Просмотр БД в виде таблицы
- Добавление новой записи
- Удаление записи (по ее номеру)
- Сохранение БД в файл
- Сортировка БД по ключевому полю (отмечено в таблице символом ‘*’)
- Поиск – выбор одной записи по ключевому полю
- Выборка – выбор группы данных соответствующих указанному признаку (по диапазону значений для поля отмеченного ‘**’).
- Решение индивидуальных задач (см. таблицу)

Программа должна быть составлена с использованием класса для описания типичного объекта, информация о котором хранится в БД.

Некоторые поля базы данных могут иметь комбинированный тип, например дата и/или время. В таких случаях рекомендуется либо использовать классы библиотеки <chrono>, либо реализовать следующие типы данных:

```
struct CTime{
    unsigned short int hour, minute, second;
};

struct CDate{
    unsigned short int year, month, day;
};

struct CDateTime{
    CDate date;
    CTime time;
};
```

Саму базу данных также необходимо разработать как класс, где внутренним скрытым полем будем массив (вектор) объектов, а методы интерфейса класса соответствуют пунктам меню программы.

Интерфейс программы может быть либо консольный (управление через повторно выводимое меню), либо оконный (с использованием элементов управления формы для визуализации данных и управления ими).

Исходный файл для тестирования БД должен содержать **не менее 25** записей.

Варианты заданий:

№ п/п	Название поля	Тип данных
1. Производители программного обеспечения		
1.	Название *	string
2.	Страна базирования	string
3.	Адрес в интернете	string
4.	Количество продуктов	unsigned int
5.	Объем продаж **	unsigned int
6.	Средняя стоимость продукции	float
Задачи		
1.	Даны два значения: целое V, вещественное – P. Для всех производителей, у которых объем продаж выше, чем V, поднимите цену на P%.	
2.	Дано название страны country. Разделите исходную базу данных на две: в первой только программное обеспечение из страны country, во второй – все остальные.	
2. Библиотека		
1.	Название книги *	string
2.	Автор	string
3.	Год издания **	unsigned short int
4.	Количество экземпляров (всего)	unsigned short int
5.	Количество экземпляров в наличии	unsigned short int
6.	Жанр	string
Задачи		
1.	Распечатайте в алфавитном порядке список авторов книг, которые пользуются популярностью у читателей, то есть для которых общее количество экземпляров больше чем в два раза превышает количество экземпляров в наличии.	
2.	Дан файл с информацией о новых поступлениях. Добавьте новые книги в БД, с учетом того, что если такая книга уже есть в БД (совпадают поля 1,2,3,6), то у нее увеличивается количество экземпляров.	

№ п/п	Название поля	Тип данных
3. Аэропорт		
1.	Номер рейса*	string
2.	Пункт прибытия	string
3.	Дата и время вылета	CDateTime
4.	Дата и время прилета	CDateTime
5.	Время в пути (вычисляемое поле)**	CTime
6.	Средняя стоимость билета	float
Задачи		
1.	В связи с погодными условиями в пункте прибытия А, все рейсы, направляющиеся в А требуется задержать на М (0 < М <1440) минут, время в пути при этом не меняется. Скорректируйте время вылета и время прилета для соответствующих рейсов.	
2.	Организуйте диалог с пользователем по подборке билетов, запросив пункт прилета, согласно нему предложите пользователю даты, на которые есть рейсы, по выбранной дате предложите сначала самый быстрый рейс, в случае отказа самый дешевый рейс, и только потом остальные варианты. Диалог должен работать пока пользователь не забронирует рейс или пока не отменит саму процедуру выбора.	
4. Железнодорожная касса		
1.	Фамилия пассажира	string
2.	Номер поезда *	unsigned short int
3.	Город отправления	string
4.	Город прибытия	string
5.	Тип вагона	char
6.	Цена билета**	float
7.	Наличие льгот	bool
Задачи		
1.	Укажите популярные для путешествий города (и отправления и прибытия). Выведите список городов, упорядоченный по количеству билетов купленных для поездки в/из данного города. Если количество билетов одинаково, города должны быть упорядочены по алфавиту.	
2.	Для заданного типа вагона есть возможность заказать платные дополнительные услуги, например питание. Это правило распространяется на всех пассажиров, кроме льготников. Перенесите в отдельную базу данных (удалив из старой) все	

№ п/п	Название поля	Тип данных
	записи для которых может быть актуальна дополнительная услуга.	
5. Автомобили		
1.	Регистрационный номер *	string
2.	Марка	string
3.	Производитель	string
4.	Год выпуска**	unsigned short int
5.	Дата постановки на учет	CDate
6.	Цвет	string
Задачи		
1.	Требуется оповестить всех владельцев машин, ставших на учет Y лет назад, о необходимости перерегистрации транспортных средств (ТС). Создайте список регистрационных номеров ТС, для владельцев которых требуется организовать соответствующую рассылку.	
2.	Дан список марок автомобилей. Перенесите в отдельную БД все автомобили заданных марок.	
6. Адресная книга		
1.	Фамилия *	string
2.	Имя	string
3.	Отчество	string
4.	Улица **	string
5.	Дом	unsigned short int
6.	Квартира	unsigned short int
7.	Телефон	string
Задачи		
1.	Сформируйте список уникальных адресов (улица, дом, квартира) и определите количество жильцов, проживающих по этим адресам.	
2.	Удалите из адресной книги всех абонентов, у которых нет телефона.	

№ п/п	Название поля	Тип данных
7. Ведомость комплектующих		
1.	Обозначение*	string
2.	Название детали	string
3.	Тип	char
4.	Цена **	float
5.	Количество	unsigned int
6.	Стоимость (вычисляемое поле)	float
Задачи		
1.	Детали различного типа требуется заказывать у различных поставщиков. Сформируйте список типов изделий и укажите для каждого типа общую стоимость изделий.	
2.	Даны две ведомости, требуется объединить их в одну таким образом, что если есть детали одного обозначения (соответственно название и тип должны совпадать), то их количество складывается, а цена указывается наибольшей из двух ведомостей.	
8. Турнирная таблица		
1.	Название команды*	string
2.	Место(определяется при сортировке по очкам)	unsigned int
3.	Количество сыгранных игр	unsigned int
4.	Выигрыши	unsigned int
5.	Ничьи	unsigned int
6.	Победы	unsigned int
7.	Очки (вычисляемое поле)**	unsigned int
Задачи		
1.	Дана информация об очередном туре в формате: указывается количество сыгранных матчей, и для каждого матча – сначала записано V, если есть победитель или D, если ничья; потом название команды победителя, потом название проигравшей команды. Если ничья, порядок следования команд не важен. Внесите результаты тура в БД.	
2.	Выведите информацию о трех лучших командах	

№ п/п	Название поля	Тип данных
9. Городской транспорт		
1.	Номер рейса*	unsigned int
2.	Тип транспорта	string
3.	Стоимость проезда**	float
4.	Время в пути	CTime
5.	Количество остановок	unsigned int
6.	Пункт отправления	string
7.	Пункт прибытия	string
Задачи		
1.	Для каждой пары пунктов (пункт отправления-пункт прибытия) укажите номер рейса с наименьшим количеством остановок.	
2.	Перенесите в новую БД информацию обо всех рейсах, время в пути которых не превышает заданное пользователем значение.	
10. Прайс-лист		
1.	Код товара*	unsigned int
2.	Название товара	string
3.	Тип	string
4.	Цена за штуку(базовая)**	float
5.	Скидка	float
6.	Цена(продажа)(вычисляемое поле)	float
Задачи		
1.	Объявлена акция на товары заданных типов, к ним дополнительно применяется скидка в р%. Измените прайс-лист с учетом акции.	
2.	Удалите из БД все товары, цена продажи которых ниже заданного порога Р.	

№ п/п	Название поля	Тип данных
11. Банковский счет		
1.	Расчетный номер*	string
2.	ФИО клиента	string
3.	Сумма на счету	float
4.	Процентная ставка	float
5.	Дата открытия счета**	CDate
6.	Привязана ли карта	bool
Задачи		
1.	Пролонгации подлежат все счета открытые более года назад от текущей даты (должна быть определена в начале работы приложения). Создайте массив клиентов, которых необходимо оповестить о пролонгации их счетов (у одного клиента может быть несколько счетов, в список он должен быть включен только один раз).	
2.	Создайте новую БД, копию текущей и пересчитайте в ней суммы на счетах на m месяцев вперед при сохранении текущих условий вклада.	
12. Расписание		
1.	Код записи (например, порядковый номер)*	int
2.	Группа	string
3.	Предмет	string
4.	Преподаватель	string
5.	День недели**	unsigned char (1..7)
6.	Номер пары	unsigned char (1..4)
Задачи		
1.	Определите, корректно ли составлено расписание: у преподавателей и у групп не может быть двух занятий на одной и той же паре. Если есть накладки в расписании, представьте информацию о них следующим образом: Группа 1, день недели, номер пары: список предметов Группа 2, день недели, номер пары: список предметов ... Преподаватель 1, день недели, номер пары: список групп Преподаватель 2, день недели, номер пары: список групп ...	
2.	Распечатайте расписание заданной группы в обще принятом виде: упорядочите по дням недели, а в дне – по номеру пары.	

№ п/п	Название поля	Тип данных
13. Оборудование		
1.	Инвентарный номер*	string
2.	Наименование	string
3.	Тип	string
4.	Дата ввода в эксплуатацию**	CDate
5.	Срок эксплуатации (месяцы, вычисляемое поле по текущей дате)	unsigned int
6.	Межповерочный интервал (месяцы)	unsigned int
Задачи		
1.	Для каждого объекта из списка оборудования, срок эксплуатации которого превышает межповерочный интервал, добавьте в начало строки с именованием символ '#’.	
2.	Удалите из списка все оборудование, срок эксплуатации которого превышает заданное значение Т. Выведите оставшиеся записи сгруппированные по типу оборудования.	
14. Острова		
1.	Название*	string
2.	Координаты	float[2]
3.	Площадь**	float
4.	Протяженность береговой линии	float
5.	Население	unsigned int
6.	Количество малых островов	unsigned int
Задачи		
1.	Разделите острова на архипелаги, то есть сгруппируйте записи таким образом, чтобы в каждой группе были только острова, расстояние (условное, вычисленное по координатам) между которым не превышает заданного значения D (для любого двух острова в архипелаге найдется другой остров из этого архипелага, такой, что расстояние между ними не более D).	
2.	Найдите m островов с наиболее протяженной береговой линией.	

№ п/п	Название поля	Тип данных
15. Страны мира		
1.	Название*	string
2.	Столица	string
3.	Население**	unsigned int
4.	Площадь	float
5.	Промышленность	string
6.	Религия	string
Задачи		
1.	Выведите информацию о странах, сгруппированную по видам промышленности. Внутри групп страны должны быть упорядочены по названию.	
2.	Создайте новую БД. Перенесите в нее те страны, плотность населения которых превышает заданный порог р.	
16. Экзаменационная ведомость		
1	Фамилия*	string
2	Имя	string
3	Отчество	string
4	Оценка (рейтинг)**	unsigned int
5	Оценка по 5-ти бальной системе (вычисляемое поле)	unsigned int
6	Допущен (да/нет)	bool
Задачи		
1.	Проверьте, правильно ли заполнена ведомость: оценка не может превышать значения в 100 баллов и не может быть выставлена оценка тому студенту, который не допущен к экзамену. Если ведомость заполнена правильно, выведите ее, упорядочив информацию в первую очередь по успеваемости, во вторую – по алфавиту. Не допущенные должны быть выделены в отдельный список.	
2.	Определите средний балл в группе. Создайте новую БД и перенесите в нее информацию обо всех студентах, успеваемость которых выше средней.	

№ п/п	Название поля	Тип данных
17. Студенческая группа		
1.	Наименование*	string
2.	Год набора**	int
3.	Баклавриат/Специалитет/Магистратура	char
4.	Факультет	string
5.	Староста	string
6.	Количество человек	int
Задачи		
1.	Выведите информацию обо всех группах заданного факультета, упорядоченную по численности студентов.	
2.	Создайте три отдельные БД для бакалавриата, специалитета и магистратуры. Перенесите в эти базы информацию о группах, исходную базу удалите.	