1. Создание модели и классов для коллекций

Постановка задачи. Необходимо создать классы для представления указанных коллекций. Реализовать стандартные операции для коллекции (добавление, удаление, поиск элемента). Реализовать интерфейс IEnumerable<T> для коллекций. Рассмотреть возможность использования нестандартного базового класса для создания коллекции.

Варианты:

* *Факультет* объединяет студенческие *группы*, а группа – *студентов*.
* *Покупка* в магазине ‑ это коллекция *товаров*. Товары в покупке могут быть в нескольких экземплярах.
* *Тест* содержит коллекцию *вопросов*, а вопрос ‑ *варианты ответов.*
* *Компьютерная сеть* содержит *компьютеры*, связанные с некими другими компьютерами в сети.
* *Матрица* – это коллекция *строк* (числовых векторов).
* *Генеалогическое дерево* содержит *людей*.

2. Простое приложение на WPF

Постановка задачи. Необходимо реализовать простое однооконное приложение WPF для просмотра и редактирования коллекции, созданной в работе 1. Применить различные элементы управления и несколько вариантов компоновки окна.

3. Связывание с данными

Постановка задачи. Необходимо реализовать отображение коллекции и отдельных ее элементов с помощью привязки данных. Воспользоваться такими возможностями, как конвертирование значений, проверка данных (при редактировании). Рассмотреть различные способы задания источника и различные режимы привязки.

4. Файлы и потоки данных

Постановка задачи. Необходимо реализовать возможность сохранения и чтения коллекции с помощью текстовых и двоичных файлов. Применить при чтении/записи потоки, поддерживающие сжатие информации (по желанию – дополнительно шифровать информацию).

5. Сериализация

Постановка задачи. Необходимо реализовать класс, позволяющий сериализовать объекты произвольных типов. Класс может производить сериализацию в любой формат данных, поддерживать один из стандартных механизмов сериализации или предоставлять новый. Применить созданный класс для сохранения данных приложения в файл.

6. LINQ to Objects

Постановка задачи. Необходимо реализовать с помощью LINQ to Objects методы для выборки из коллекции, упорядочения и группировки элементов коллекции, вычисления агрегатных функций. Воспользоваться LINQ-запросами для отображения в приложении необходимой части коллекции. По желанию сделать свою реализацию оператора упорядочения, группировки, объединения или работы с множествами, применяя концепцию отложенных вычислений там, где это возможно.

7. Многопоточное программирование

*(Данная лабораторная работа делается отдельно от остальных.)*

Постановка задачи. Необходимо создать класс для представления музыкальной композиции со следующими атрибутами: ID (уникальный); название (строка длиной до 256 символов); длина (минуты, секунды); исполнитель (строка длиной до 256 символов); жанр (значение из перечисления допустимых жанров); рейтинг (целое число от 0 до 10). Создать класс для представления плей-листа, представляющего коллекцию неповторяющихся композиций и обладающего следующими элементами: ID (уникальный); название (строка длиной до 256 символов); общая длина (минуты, секунды); рейтинг (высчитывается как среднее рейтинга композиций). Обеспечить хранение плей-листа в файле (в произвольном формате). Создать «плеер» для загрузки и «проигрывания» плей-листов. Проигрывание должно происходить в отдельном потоке, при этом допускается одновременное проигрывание нескольких плей-листов. Каждый загруженный плей-лист должен отображаться на отдельной вкладке формы. Проигрыванием можно управлять с помощью элементов управления окна (пауза/стоп). При «проигрывании» композиции выводятся текстовые метки или звуковые сигналы для каждых 5 секунд проигрывания.

8. Ресурсы и локализация

Постановка задачи. Необходимо дополнить формы проекта (как минимум главную) меню, предоставляющим доступ к наиболее часто выполняемым операциям. Меню должно предусматривать горячие клавиши и пиктограммы. Для хранения пиктограмм использовать ресурсы приложения. С помощью стандартных ресурсных файлов интернационализировать приложение (названия пунктов меню, всплывающих подсказок, заголовок окна и т. п., в том числе хотя бы одну картинку), предоставив хотя бы две локализации. Обеспечить сохранение выбранного пользователем языка в настройках или возможность изменения языка «на лету» (без перезапуска приложения).

9. Графика и анимация

Постановка задачи. Необходимо создать визуальный стиль для некоторых (минимум двух) элементов управления. Создать шаблон для выбранного элемента управления или элемента данных. Создать в проекте анимации для элементов управления (эффект «свечения», плавного изменения размеров при выделении и т.п.). Использовать несколько видов анимации (с использованием линейной интерполяции, ключевых кадров, пути). Применить в проекте работу с картинками. По желанию освоить основы работы с аудио или видео в WPF.