Курсовые по ПЯВУ (ООП)

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.

2. Реализация задания в консоли или графике.

3. Работа с файлами.

Виды коллекций:

1. Списки (1, 2 связные) и стек на их основе
2. Коллекция ключ-значение (с уникальными или неуникальными ключами)
3. Список или стек на основе расширяющегося массива

Примечание

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

**Курсовой по ООП. Вариант 1**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу для моделирования почтовых сортировочных центров. С самого крупного (единственного) центра посылки следуют, в зависимости от индекса, в более мелкие центры, далее в еще более мелкие центры и т.д. Чем крупнее центр, тем больше нулей в последних цифрах его индекса. В программе должно быть не менее 3-х уровней сортировочных центров. Список сортировочных центров с их индексами хранится в файле конфигурации программы. Данные о посылках: адрес отправителя, адрес получателя, индекс, вес, считываются из файла. Программа должна пошагово смоделировать работу центров. Использовать класс «очередь» на основе односвязного списка.

**Курсовой по ООП. Вариант 2**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу, отыскивающую проход по лабиринту, с использованием класса «стек» на основе двусвязного списка. Лабиринт представляется в виде матрицы, состоящей из квадратов. Каждый квадрат либо открыт, либо закрыт. Вход в закрытый квадрат запрещен. Если квадрат открыт, то вход в него возможен со стороны, но не с угла. Программа находит проход через лабиринт, двигаясь от заданного входа. После отыскания прохода программа выводит найденный путь в виде координат квадратов. Карта считывается из файла вместе с координатой входа в лабиринт. Программа должна в пошаговом режиме продемонстрировать поиск выхода.

**Курсовой по ООП. Вариант 3**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу, моделирующую управление каталогом в файловой системе. Для каждого файла в каталоге содержатся следующие сведения: имя файла, дата создания, количество обращений к файлу.

Программа должна обеспечивать:

* создание каталога или файла в выбранном каталоге;
* вывод содержимого каталога (аналог команды dir в cmd);
* удаление файлов, дата создания которых раньше заданной;
* просмотр и редактирование файла;
* выборку файла с наибольшим количеством просмотров;

Выбор моделируемой функции должен осуществляться с помощью меню. Для представления каталога использовать класс «двусвязный список». Хранить всю информацию (всю систему каталогов и файлов) в базе данных в виде текстового файла.

**Курсовой по ООП. Вариант 4**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу моделирования работы автобусного парка. Сведения о каждом автобусе содержат: номер автобуса, фамилию и инициалы водителя, номер маршрута. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* формирование данных об автобусах в парке;
* имитация выезда автобуса из парка: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся в парке, и записывает эти данные в список автобусов, находящихся на маршруте;
* имитация въезда автобуса в парк: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся на маршруте, и записывает эти данные в список автобусов, находящихся в парке;
* вывод сведений об автобусах, находящихся в парке, и об автобусах, находящихся на маршруте.

Для представления необходимых списков использовать класс «двусвязный список». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 5**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу учета заявок на авиабилеты. Каждая заявка содержит: пункт назначения, номер рейса, фамилию и инициалы пассажира, желаемую дату вылета. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* добавление заявок в список;
* удаление заявок;
* вывод заявок по заданному номеру рейса и дате вылета;
* вывод всех заявок.

Для хранения данных использовать класс «двусвязный список». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 6**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу учета книг в библиотеке. Сведения о книгах содержат: фамилию и инициалы автора, название, год издания, количество экземпляров данной книги в библиотеке.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* добавление данных о книгах, вновь поступающих в библиотеку;
* удаление данных о списываемых книгах;
* выдача сведений о всех книгах, упорядоченных по фамилиям авторов;
* выдача сведений о всех книгах, упорядоченных по годам издания.

Хранение данных организовать с применением класса «коллекция ключ-значение» (с возможностью хранить данные с одинаковыми ключами), в качестве ключа использовать «фамилию и инициалы автора». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 7**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу «Моя записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по разным полям. Хранение данных организовать с применением класса «коллекция ключ-значение», с ее же помощью организовать хранение признаков (ключ) и ссылок на сообщения (значение). Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 8**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу учета заявок на обмен квартир и поиска вариантов обмена. Каждая заявка содержит сведения о двух квартирах: требуемой (искомой) и имеющейся. Сведения о каждой квартире содержат: количество комнат, площадь, этаж, район.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* ввод заявки на обмен;
* поиск в картотеке подходящего варианта: при совпадении требований и предложений по количеству комнат и этажности и различии по показателю «площадь» в пределах 10% выводится соответствующая карточка и удаляется из списка, в противном случае поступившая заявка включается в картотеку;
* вывод всей картотеки.

Для хранения данных картотеки использовать класс «двусвязный список». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 9**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу «Автоматизированная информационная система на железнодорожном вокзале». Информационная система содержит сведения об отправлении поездов дальнего следования. Для каждого поезда указывается: номер поезда, станция назначения, время отправления.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* ввод данных в информационную систему;
* вывод сведений по всем поездам;
* вывод сведений по поезду с запрошенным номером;
* вывод сведений по тем поездам, которые следуют до запрошенной станции назначения.

Хранение данных организовать с применением класса «двусвязный список». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 10**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу «Англо-русский и русско-английский словарь». «База данных» словаря должна содержать синонимичные варианты перевода слов. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* Загрузка «базы данных» словаря (из файла).
* Выбор режима работы:
  + англо-русский;
  + русско-английский.
* Вывод вариантов перевода заданного английского слова.
* Вывод вариантов перевода заданного русского слова.

Базу данных словаря реализовать в виде двух контейнеров типа «коллекция ключ-значение».

**Курсовой по ООП. Вариант 11**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу, реализующую игру «Крестики-нолики» между двумя игроками: пользователем и компьютером (роботом). В программе использовать двусвязные списки.

**Курсовой по ООП. Вариант 12**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу, решающую игру-головоломку «Игра в 15». Начальное размещение номеров — случайное. Предусмотреть два режима демонстрации решения: непрерывный (с некоторой задержкой визуализации) и пошаговый (по нажатию любой клавиши). В программе использовать двусвязные списки.

**Курсовой по ООП. Вариант 13**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Составить программу формирования списка кандидатов, участвующих в выборах губернатора. Каждая заявка от кандидата содержит: фамилию и инициалы, дату рождения, место рождения, индекс популярности.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* Добавление заявки в список кандидатов. Для ввода индекса популярности (значение указано в скобках) предусмотреть выбор с помощью подменю одного из следующих вариантов:
  + поддержан президентом (70);
  + поддержан оппозиционной партией (15);
  + оппозиционный кандидат, который снимет свою кандидатуру в пользу кандидата № 1 (10);
  + прочие (5).
* Удаление заявки по заявлению кандидата.
* Формирование и вывод списка для голосования.

Хранение данных организовать с применением класса «очередь с приоритетом». Для надлежащего функционирования очереди с приоритетами побеспокоиться о надлежащем определении операции < (меньше) в классе, описывающем заявку кандидата. Формирование и вывод списка для голосования реализовать посредством выборки заявок из очереди.

**Курсовой по ООП. Вариант 14**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу моделирования работы автобусного парка. Сведения о каждом автобусе содержат: номер автобуса, фамилию и инициалы водителя, номер маршрута. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* формирование данных об автобусах в парке;
* имитация выезда автобуса из парка: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся в парке, и записывает эти данные в список автобусов, находящихся на маршруте;
* имитация въезда автобуса в парк: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся на маршруте, и записывает эти данные в список автобусов, находящихся в парке;
* вывод сведений об автобусах, находящихся в парке, и об автобусах, находящихся на маршруте.

Для представления необходимых списков использовать класс «коллекция ключ-значение», в качестве ключа использовать «номер автобуса». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 15**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу учета заявок на авиабилеты. Каждая заявка содержит: пункт назначения, номер рейса, фамилию и инициалы пассажира, желаемую дату вылета. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* добавление заявок в список;
* удаление заявок;
* вывод заявок по заданному номеру рейса и дате вылета;
* вывод всех заявок.

Для хранения данных использовать класс «коллекция ключ-значение» (с неуникальными ключами), в качестве ключа использовать «пункт назначения». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 16**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу учета книг в библиотеке. Сведения о книгах содержат: фамилию и инициалы автора, название, год издания, количество экземпляров данной книги в библиотеке.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* добавление данных о книгах, вновь поступающих в библиотеку;
* удаление данных о списываемых книгах;
* выдача сведений о всех книгах, упорядоченных по фамилиям авторов;
* выдача сведений о всех книгах, упорядоченных по годам издания.

Хранение данных организовать с применением класса «список» (на основе расширяющегося массива). Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 17**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу «Моя записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по разным полям. Хранение данных организовать с применением класса «двусвязный список». Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.

**Курсовой по ООП. Вариант 18**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу для моделирования почтовых сортировочных центров. С самого крупного (единственного) центра посылки следуют, в зависимости от индекса, в более мелкие центры, далее в еще более мелкие центры и т.д. Чем крупнее центр, тем больше нулей в последних цифрах его индекса. В программе должно быть не менее 3-х уровней сортировочных центров. Список сортировочных центров с их индексами хранится в файле конфигурации программы. Данные о посылках: адрес отправителя, адрес получателя, индекс, вес, приоритет посылки, считываются из файла. Программа должна пошагово смоделировать работу центров. Использовать класс «очередь с приоритетами» на основе массива списков (массив – приоритеты, элемент массива – односвязный список).

**Курсовой по ООП. Вариант 19**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу, отыскивающую проход по лабиринту, с использованием класса «стек» на основе расширяющегося массива. Лабиринт представляется в виде матрицы, состоящей из квадратов. Каждый квадрат либо открыт, либо закрыт. Вход в закрытый квадрат запрещен. Если квадрат открыт, то вход в него возможен со стороны, но не с угла. Программа находит проход через лабиринт, двигаясь от заданного входа. После отыскания прохода программа выводит найденный путь в виде координат квадратов. Карта считывается из файла вместе с координатой входа в лабиринт. Программа должна в пошаговом режиме продемонстрировать поиск выхода.

**Курсовой по ООП. Вариант 20**

Курсовой нацелен на изучение и закрепление следующих тем:

1. Реализация универсальной объектной коллекции (шаблона, template), указанной в задании.
2. Реализация задания в консоли или графике.
3. Работа с файлами.

**Примечание**

В курсовом необходимо написать УНИВЕРСАЛЬНЫЙ класс объектной коллекции, который сможет работать с ЛЮБЫМ типом данных (см. лекции раздел «Обобщенное программирование»). И затем этот класс нужно использовать в программе в каком-то частном применении с явным указанием типов данных, с которым он работает.

Для контроля студента со стороны преподавателя, а также самоконтроля студента по корректной с точки зрения требований курсового проекта реализации объектной коллекции (шаблона), в пояснительной записке на уровне листинга исходного кода продемонстрировать использование шаблона с отличным от заданного в курсовом проекте типом параметра. Например, продемонстрировать работу коллекции с элементарным типом данных int, ибо корректно реализованный шаблон должен поддерживать любые типы данных в качестве параметра.

Внутри классов не должно быть никакой работы с интерфейсом, т.к. интерфейс от программы к программе может меняться. Исключения: специальные методы вывода класса на экран (Print) и специальные интерфейсные классы, которые созданы с одной-единственной целью - работать с интерфейсом (вдруг кто-то захочет создать класс Menu, который отображает и обрабатывает работу в консоли с менюшкой, - так можно).

**Задание**

Написать программу «Автоматизированная информационная система на железнодорожном вокзале». Информационная система содержит сведения об отправлении поездов дальнего следования. Для каждого поезда указывается: номер поезда, станция назначения, время отправления.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

* ввод данных в информационную систему;
* вывод сведений по всем поездам;
* вывод сведений по поезду с запрошенным номером;
* вывод сведений по тем поездам, которые следуют до запрошенной станции назначения.

Хранение данных организовать с применением класса «коллекция ключ-значение» (с неуникальными ключами), в качестве ключа использовать номер поезда. Предусмотреть сохранение всех данных при выходе в файл и восстановление при повторном запуске программы.