

«Платформа Microsoft .NET и язык программирования C#»

Встреча №4

ТЕМА: ИНДЕКСАТОРЫ

Цель: Совершенствование навыков применения объектно-ориентированного подхода в программировании с использованием средств C#, создания пользовательских типов, использования средств обработки исключительных ситуаций.

Необходимые инструменты: MS Visual Studio 2016.

Документация: Конспект, Литература.

Ориентировочное время исполнения: 1 час.

Задание 1.

В C # индексация начинается с нуля, но в некоторых языках программирования это не так. Например, в Turbo Pascal индексация массиве начинается с 1. Напишите класс *RangeOfArray*, который позволяет работать с массивом такого типа, в котором индексный диапазон устанавливается пользователем. Например, в диапазоне от 6 до 10, или от -9 до 15.

Подсказка: В классе можно объявить две переменных, которые будут содержать верхний и нижний индекс допустимого диапазона.

«Платформа Microsoft .NET и язык программирования C#»

Встреча №5

ТЕМА: НАСЛЕДОВАНИЕ

Цель: Закрепить у слушателей практические навыки и теоретические знания для работы классами и объектами, свойствами. Научиться создавать иерархии классов.

Необходимые инструменты: MS Visual Studio 2016.

Документация: Конспект, Литература.

Ориентировочное время исполнения: 2 часа.

Требования к отчету: Отчет должен быть оформлен в виде электронного документа: программный код с комментариями, выводы о результатах выполняемых действий и копии экрана. Размер файла отчета до 2 МБ со скриншотами.

Задание 1. Разработать приложение «Резервная копия»

Цель: произвести расчет необходимого количества внешних носителей информации при переносе за один раз важной информации (565 Гб, файлы по 780 Мб) с рабочего компьютера на домашний компьютер и затрачиваемое на данный процесс время. Вы имеете в распоряжении следующие типы носителей информации:

- Flash-память,
- DVD-диск,
- съемный HDD.

Каждый носитель информации является объектом соответствующего класса:

- Flash-память — класс «Flash»;
- класс DVD-диск — класс «DVD»;
- класс съемный HDD — класс «HDD».

Все три класса являются производными от абстрактного класса «Носитель информации» — класс «Storage».

Базовый класс («Storage») содержит следующие закрытые поля:

- наименование носителя;
- модель.

Класс обладает всеми необходимыми свойствами для доступа к полям, а также абстрактными методами:

- получение объема памяти;
- копирование данных (файлов/папок) на устройство,
- получение информации о свободном объеме памяти на устройстве;
- получение общей/полной информации об устройстве.

Кроме того, каждый из производных классов дополняется следующими полями:

- класс Flash-память: скорость USB 3.0, объем памяти;
- класс DVD-диск: скорость чтения / записи, тип (односторонний (4.7 Гб) /двусторонний (9 Гб));
- класс съемный HDD: скорость USB 2.0, количество разделов, объем разделов.

Работа с объектами соответствующих классов производится через ссылки на базовый класс («Storage»), которые хранятся в массиве.

Приложение должно предоставлять следующие возможности:

- расчет общего количества памяти всех устройств;
- копирование информации на устройства;
- расчет времени необходимого для копирования;
- расчет необходимого количества носителей информации представленных типов для переноса информации.

«Платформа Microsoft .NET и язык программирования C#»

Встреча №6

ТЕМА: СТРУКТУРЫ И ПЕРЕЧИСЛЕНИЯ

Цель: Закрепить у слушателей практические навыки и теоретические знания для работы классами и объектами, свойствами. Научиться использовать структуры и перечисления.

Необходимые инструменты: MS Visual Studio 2016.

Документация: Конспект, Литература.

Ориентировочное время исполнения: 2 часа.

Требования к отчету: Отчет должен быть оформлен в виде электронного документа: программный код с комментариями, выводы о результатах выполняемых действий и копии экрана. Размер файла отчета до 2 МБ со скриншотами.

Задание 1. Приложение «Фигуры»

Написать приложение, которое будет отображать в консоли выбранные пользователем простейшие геометрические фигуры: «прямоугольник», «ромб», «треугольник», «трапеция», «многоугольник». В меню пользователь выбирает фигуры и выбирает цвета для каждой фигуры. Фигуры рисуются звездочками или другими символами. Для реализации программы необходимо разработать иерархию классов (продумать возможность абстрагирования). Все заданные пользователем фигуры хранятся в объекте «Обобщённая фигура», который имеет метод «отобразить все выбранные фигуры».