Лабораторная работа №14

**«Классы. Множественное наследование, виртуальные функции и абстрактные классы»**

1. **Пример множественного наследования**. Создать производный класс, описывающий окружность, вписанную в квадрат. Базовые классы – окружность и квадрат **(lab14\_1.cpp)**
2. Множественное наследование с дублированием базового класса **(lab14\_2.cpp)**
3. Одинаковые функции в базовом и производном классах (Выбор функции зависит только от типа указателя, но не от его значения) **(lab14\_3.cpp)**
4. Виртуальная функция в базовом классе (Выбор экземпляра функции выбирается объектом, то есть выбор зависит от значения указателя – типа объекта, для которого выполняется вызов) **(lab14\_4.cpp)**
5. Некоторые особенности виртуальных функций – при подмене виртуальной функции требуется полное совпадение сигнатур имен и типов возвращаемых значений функций в базовом и производном классах **(lab14\_5.cpp)**
6. Механизм виртуального вызова может быть подавлен с помощью явного использования полного квалифицированного имени **(lab14\_6.cpp)**
7. Абстрактный класс на базе класса (где есть хотя бы одна чистая виртуальная функция) **(lab14\_7.cpp)**
8. Внешние, локальные и глобальные классы **(lab14\_8.cpp)**
9. **Задание.** Создать производный класс "Аэроплан", для которого базовыми являются классы "Транспортное средство" и "Компьютер". В классе "Компьютер", который описывает бортовой компьютер самолета, прописать защищенное поле baseTimeToBoot – время загрузки компьютера – и методы GetTimeToBoot () – возвращает время загрузки компьютера, обращения к этим полям. Вне класса задать поле значения времени загрузки компьютера по умолчанию DefaultTimeToBoot , в конструкторе класса "Компьютер" задавать для поля baseTimeToBoot значения DefaultTimeToBoot, если не вводится никакое другое значение. В классе "Аэроплан" выполнить обновление бортового компьютера с помощью метода ComputerUpdate (double newTimeToBoot).
10. **Задание.** Создать абстрактный базовый класс "IWhoAmI" с чистой функцией Iam(), от которого производным будет класс "ФИО", содержащий конструктор, деструктор, поля "имя" и "возраст", методы ввода возраста, имени, получения возраста и имени. Создать класс "Человек", производный от класса "ФИО", с конструктором, деструктором, методом ввода номера паспорта, методом получения номера паспорта и с пустой виртуальной функцией Iam(), которая будет выводить на экран надпись "I a man object", от которого создать производные классы "Студент", "Пенсионер". Для этих производных классов сделать базовый класс "Человек" – виртуальным. Создать производный класс "Сотрудник", для которого базовыми классами будут "Студент" и "Пенсионер". Для каждого из производных классов функция Iam() будет выводить разные надписи: "Я студент", "Я сотрудник", "Я пенсионер". В функции void main() обеспечить выбор создания и вывода объекта того или иного класса и обеспечить выполнение функции Iam() для выбранного класса.