

1. Создать три перечисления:
  - DVDROMState со значениями empty и withDisk. Для последнего варианта добавить возможность хранить ассоциативное значение в виде строки - названия программы/игры на диске.
  - PowerState со значениями on, off.
  - MonitorState со значениями on, off, sleep, screenSaver.
2. Описать структуру PeripheralDevice со свойством name типа строка.
3. Описать класс Computer для которого задать свойства типов перечислений, указанных выше и массив подключенных девайсов (структуры, описанные по 2 заданию). Придумать свои дополнительные свойства (2-3 достаточно), которые должны быть у типового компьютера.
4. Описать для класса Computer методы для взаимодействия со свойствами:
  - turnOn и turnOff для корректного включения/выключения компьютера.
  - changeScreenState для изменения состояния монитора.
  - addDevice и removeDevice для управления периферией.  
*Дополнительно\*: использовать для управления периферией subscript.*
  - insertDisk и takeOutDisk для возможности вставить диск с программой/игрой
5. Добавить в эти методы некоторую проверку (например нельзя включить питание, пока не подключен монитор или блок питания).
6. Описать для класса Computer классы наследники Notebook и Supercomputer. Подумать, что может быть отличного у этих классов от типового компьютера, добавить 1-2 дополнительных свойства, которые могут быть только у этих классов, но не у класса Computer.
7. Переопределить 1-2 метода класса родителя в любом классе-наследнике, добавить новые методы, специфичные для данного класса. Подумать, возможно какие-то свойства/методы родителя будут не актуальны для наследника.  
*Подсказка: у ноутбука может не быть DVD привода.*
8. Создать несколько экземпляров каждого из классов и провести с ними некоторые манипуляции.
9. Добавить поддержку классами протокола CustomStringConvertible и вывести информацию по каждому из созданных объектов в консоль.
10. Добавить протокол DefaultComputerProtocol с типовыми свойствами и методами простого компьютера. Использовать данный протокол для описанных выше классов.
11. Добавить в расширении протокола дефолтную реализацию указанных в протоколе методов. Избавиться от наследования между классами, разделив их на 3 независимых класса, имплементирующих протокол DefaultComputerProtocol.
12. Создать массив объектов, описанных выше классов и написать функцию openComputerShop, принимающую этот массив и включающую все компьютеры из него.  
*Пояснение: Как будто у нас в компьютерном магазине стоит много компьютерной техники, и мы в начале рабочего дня хотим одним рубильником запустить их все.*
13. Подумать над реализацией такого же метода, но для конца рабочего дня (выключить всё).