Inheritance Worksheet (Wednesday)

A. Design Problem

The following is a description of two classes associated with this design problem.

1. DVDPlayer

   a. Data Members
      
      offOn - tells whether the DVD is on or off  
      trackNum - number of current track

   b. Methods
      
      - stop - stops playing  
      - play - continues playing at current track
      - pause - stop playing staying at current track
      - eject - eject the DVD
      - nextTrack - moves forward by one track
      - prevTrack - moves backward by one track
      - selectSubMenu - selects a menu from the screen
      - turnOnOff - turns DVD on or off

2. VCR

   a. Data Members
      
      offOn - tells whether the VCR is on or off

   b. Methods
      
      - stop - stops playing  
      - play - continues playing at current tape location
      - pause - stop playing staying at current tape location
      - eject - eject the tape
      - rewind - rewinds the tape
      - record - starts recording
      - SpEpMode - switches between SP and EP mode.
      - medium - returns the recording medium
      - capacity - returns the recording capacity (in hours)
      - turnOnOff - turns VCR on or off
3. **What you must implement**

You must define a set of classes and an interface which satisfy the following requirements:

a. You must avoid code duplication in your hierarchy.

b. You must define a base class that allow us to pass to a method either a DVDPlayer or a VCR. Pick a name for the base class that reflects the appropriate functionality.

c. We consider the VCR a recording device. Define an interface (named RecordingDevice) which allow us to treat a VCR as a recording device.

d. You just need to provide a diagram showing the classes you define, the interface and the relationship between them.

B. Using your design write the code implementing the classes/interface you defined above. Feel free to define parameters and return types for methods as you understand are necessary.

C. Write a class that illustrates how to define and use polymorphic references using the previously defined class(es) and interface(s).