

# Introduction to Computers and Engineering Problem Solving 1.00 / 1.001 Fall 2005

---

## Problem Set 4

Due: 11AM, Friday October 14, 2005

## TIVO 1 [100 points]

### Introduction

In this problem set, we will continue to build our TIVO system.

You may have noticed in the previous problem set that there is a good bit of overlap in all the TV program classes. It is redundant to have a "title" field, for example, in each of the different program classes, when that is clearly a field that they will all share. Now that you have learned about inheritance, you know that in cases like this, it makes sense to have a class that all the program classes extend. This also allows us to use the super class as a type (instead of "Object") when we know we are dealing with a TV program, but we don't know exactly which kind.

You will extend the TIVO system further by adding a Recordings class. This will correspond to the "WishList" in real life TIVO. You will need a Recording class as well which will represent an object to be recorded. It is not enough to use a TV Program object because it does not contain information about time and channel.

We give you a TIVO class which is the container for the TIVO system that we are building. Make sure to read this class before attempting the problem set.

### Assignment

Create or modify the following:

`Movie, News, Series, Special`

- Determine how to change each of these classes.

`TVProgram`

- Determine what should go into this class. Should it be abstract or concrete?

`DailySchedule`

- If the above changes call for changes in this class, make sure that you take care of that.
- Write the following methods:
  - `channelNameToNumber(String name)`
    - Takes a channel in String form and returns the int version. To find the int version of a channel, take the channels array (which is a static final data

member of the DailySchedule class) and return the position of the String version of the channel. If there is no corresponding position, return Integer.MAX\_VALUE

- o `getTVProgram(String channel, int timeslot)`
  - Takes a channel and a timeslot and returns the TVProgram that is running at that time.

#### Recording

- Create the Recording class, which has the following data members: TVProgram `tvp`, Calendar date, int timeslot, String channel. Decide what modifiers these variables have and what constructor(s) to create. If appropriate, create get and set methods. Make sure to write a constructor.

#### Recordings

- Create the Recordings class. This class has the following data members:
  - o `public static final int NUM_RECORDING_SLOTS` (see the TIVO class data members for a hint of what this should be set equal to)
  - o `private int memoryUsed`; (this represents the space used on the hard drive)
  - o `private ArrayList recordings`;
  - o `private DailySchedule schedules[]`;
- Create a constructor which takes an array of DailySchedule and initializes all the data members.
- Create `get()` and `set()` methods for the data members.
- Write the following methods:
  - o `existsConflict(Recording toAdd), existsConflict(Calendar date, int timeSlot)`
    - takes a Recording or a date and timeslot and checks the recordings ArrayList to see if another recording is already scheduled to be recorded at that time. Returns true if there is a conflict and false otherwise.
  - o `isSpace()`
    - checks to see if there is space on the hard drive to record another program. You can assume that every program is half an hour. Returns true if there is space for another program and false otherwise.
  - o `record(String channel, Calendar date, int timeSlot), record(Recording toAdd)`
    - takes a Recording or a channel, date, and timeslot and adds the program to the Recordings ArrayList. Don't forget to check for conflict, make sure that there is room on the hard drive, and update hard drive (memory used) to reflect the addition of this recording. Returns true if the recording was added successfully and false otherwise.
  - o `recordAll(String title)`
    - records all programs in the schedule with the specified title. Do not worry about conflicts – assume this takes precedence over previously scheduled recordings.

Returns true if the recordings were added successfully and false otherwise.

- `remove(Recording toRemove)`
  - takes a `Recording` and removes the program from the `Recordings ArrayList`. Don't forget to update the hard drive (memory used) to reflect the removal of this recording. Returns true if the recording was removed successfully and false otherwise.
- `remove(Calendar date, int timeslot)`
  - takes a date and timeslot and removes the corresponding `Recording` from the `Recordings ArrayList`. Don't forget to update the hard drive (memory used) to reflect the removal of this recording. Returns the `Recording` that was removed.
- `remove(String title)`
  - removes all programs in the recordings with the specified title. Returns an `int`: the number of recordings that were removed.
- `print()`
  - prints a list of the scheduled recordings. For each recording, you should print the title of the program and the date and time that it is being recorded.

## Turn In

- Turn in **electronic** copies of **all source code** (`.java` files). No printed copies are required.
- Place a **comment** with your full name, MIT server username, tutorial section, TA's name, and assignment number at the beginning of all `.java` files in your solution.
- Remember to **comment your code**. Points will be taken off for insufficient comments.
- Place all of the files in your solution into a **single zip file**. Submit this single zip file under the appropriate section and problem set number.
- Your solution is **due at 11AM**. Your uploaded files should have a time stamp of no later than 11AM on the due date.
- **Do not** turn in compiled byte code (`.class` files) or backup source code (`.java~` files).

## Penalties

- **30% off** If you turn in your problem set **after 11AM on Friday** but **before 11AM on the following Monday**.
- **No Credit** If you turn in your problem set **after 11AM on the following Monday**.