Student Name _____

Your Temporary Group Number

TG1   TG2   TG3   TG4   TG5   TG6

TG7   TG8   TG9   TG10   TG11   TG12

Name of student who is recording:_____

Name of student who will present for group if called on:

_____

Assume that you are able to construct a web application that solves Prof. Hilfbar's scheduling problem.

Somewhere on the web server, a piece of software will take the stuff entered by Prof. Hilfbar and the students, and turn it into the answer that Prof. Hilfbar is seeking.   For now, it is enough to know that this piece of software will be written in Java (we'll talk about how it fits in with the rest of the system later.)

Given that this piece of software will be written in Java, you are invited to think about what classes/objects you might need.

Sketch out the specification for a few classes and methods that you'll need for this application. At least one of these classes should have a method that returns an object which is some representation of the "answer to the problem"; something object containing data that, if properly formatted in HTML, could be the stuff that comes up on the screen to tell Prof. Hilfbar the solution to his scheduling problem.

Sketch out your classes and methods on transparencies, or on a laptop in either Powerpoint or an ASCII file that could be posted to the web. Be prepared to present them to the class when time is called.

If there is anything you *don't know* that you need to know in order to design these classes—about Java, or about Prof. Hilfbar's problem, about HTML, about web applications based on Java, or anything else—make a note of it.   In the Problem Based Learning (PBL) framework, these things you "don't know" that you "need to know" are called *learning issues*.   Just list them for now. We'll do something with that list at a later stage.