Scheduling

4/17/2009

Opening Discussion

- Minute Essay comments
  - There are different approaches to scheduling, but some basic things to consider.
    - Required classes
    - Desirability
    - Frequency of offering
    - Conflicting times
    - Consecutive classes

The Problem

- We want to write a program that will find optimal schedules for us.
- Doing full class schedules is hard so we will make some simplifying assumptions.
  - Instead of using times, we will use blocks.
  - Have to have some way of handling requirements.
  - Give classes a weight based on value to us.
  - Only print schedules above a certain weight.

The Approach

- We will put the list of possible classes in a spreadsheet.
- We will use recursion to test all possible configurations.
  - The main argument will be the index class we are considering.
  - Other arguments will keep track of other information for us.
  - The base case is when we get past the last class.

More Approach

- Recursive cases
  - We always call on the next class without picking this one.
  - If this one is needed/allowed, we also recurse having picked this one.

Factors to Consider

- Let's have a discussion of what factors we can consider.
- Under what condition do we not want to schedule a class?

Code

- Let's start writing the recursive function to handle this.
- What types do we want for the extra arguments that help us determine if we can add a class on top of a specific schedule?

Minute Essay

- What questions do you have about today's topic? Can you think of what we would need to do to get around some of the limitations we put into our code?
- I'm thinking that after this topic we will look into applets and “normal” Java programming.