

Computer Science 250: Final Project
Andreas Bach Landgrebe
Lucas Hawk
Viktor Zheng
Monday April 13, 2015
Progress Report 2

As of so far, we have made a good amount of progress towards the algorithms that we have set out to implement. Our project consists of creating a class scheduling algorithms using GAs(Genetic Algorithms). Since the last time, we have decided to set up SAX parsing. Doing so will allow us to populate arrays of students or professors or any other objects that have called from an XML file. We have decided to do a SAX parsing compared to a DOM parsing for a number of reasons. One of these reasons include that we do not need continuous access to these data once we have populate the arrays of students or professors or any other objects. The SAX parsing also has a large space overhead compared to the DOM parsing so we do not need to worry about how many students need to schedule of specific classes. Having this worry to not be a concern will allow us to suggest such a scheduling system to a large university. This is one of reasons that our class schedule algorithm will be better compared to another.

One thought that we had since the past progress report is considering how we plan to evaluate our system. The way we plan to evaluate this system is by using the following evaluation metrics; preferences of students and requirements.

When considering the evaluation metric, preferences of students, there are a general focus to have. This focus includes looking at what classes students prefer to take and make sure that this class scheduling will not violate the student from a specific class. Students will have preferences to be able to take a specific class that is being offered at a given semester. Our algorithm will ensure that students will be able to schedule this classes that students would want to take.

The other evaluation metric that we will focus on is the requirements. These requirements include having students register for classes that do not conflict that one another. For example, if a computer science class is being offered Monday, Wednesday and Friday from 10AM to 11AM, then our scheduling algorithm will

make sure that a class such as a Communication Arts class will not be scheduled when it is offered Monday, Wednesday and Friday from 10:30AM to 11:15AM. It is important for the metric to be in place in order to make sure a student does not have to be in two places at once.

In this system, we have 7 java source code algorithms. Class.java has different objects to be used to organize this system the most effective way. In this file, we also get and setters and return these objects to get information about the professor, number of students already enrolled, the information about a course, and classroom that these classes will be held at.

The next file, Classroom.java layouts another part of the class scheduling system. This java file has getter and setter method to be able to return the ID of a student, get the name of a student, and also return the number of seats that a class has to offer.

Course.java is yet another file to be used for the design of this project. This will return the ID and the name of the specific course offered.

Professor.java will include yet more objects to be used. This java file will return the information about objects for the ID of the professor, the classes that the professor is teaching, the maximum number of classes that the professor could possibly teach, and the current amount of classes that a professor is teaching.

The next java file that is in the designed system is Schedule.java. This java file is meant to have students be able to organize their schedule by knowing the specific times that courses are being offered. This will have students be able to add and remove classes from their schedule given the time that it is being offered. We have also added the fitness to determine how well this schedule will work with a student. We still have to discuss how we will write this algorithm to determine how well a proposed schedule will work with a student.

Student.java will return a list of courses that a student is taking.

The last java file in the current proposed project is TimeTabler.java. This file is our main file to include a number of ArrayList to present all of the information from the objects that have been declared in the past files.

So far, we have organized a system to use genetic algorithms to determine a class schedule between a professor and a student. There is great amount of programming to be done in order to complete this proposed project.