Prerequisite Validator

Tawan Scott

15 May 2020

Abstract

This paper will go into detail about the DFA project and its applicational purposes. This includes an introduction of the project, detailed system description, requirements for the application, literature survey that reviews other similar applications/systems, user manual that guides the users, and a DFA diagram.

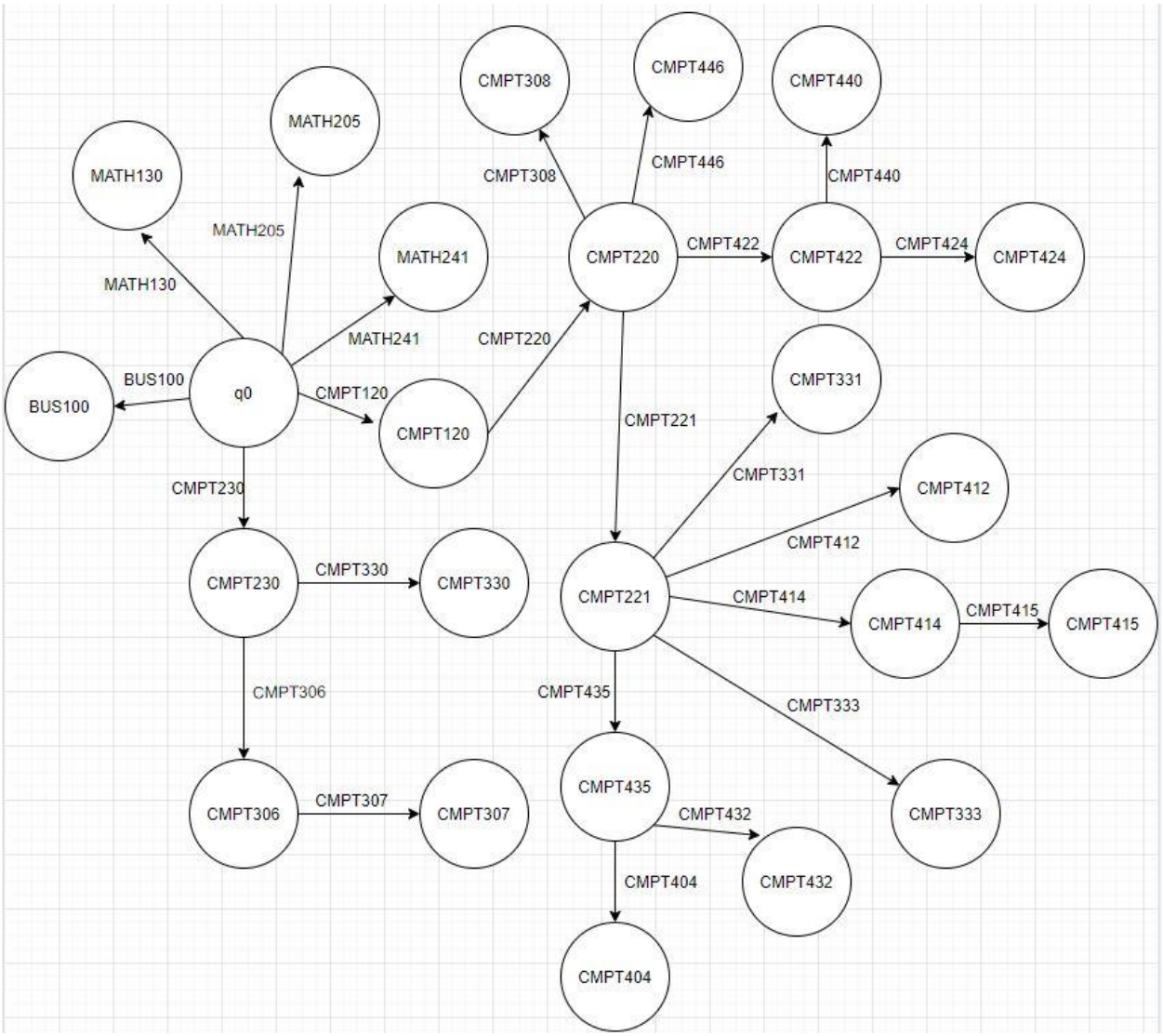
Introduction

Class registration can be a very stressful period for students, especially when there are so many high-level courses that require many different prerequisites. Without a proper application or system that will help students figure out if they had met their prerequisites requirements, registration can become a huge mess. Prerequisite Validator will help define whether or not a computer science student has met the requirements for a course that they are planning to take.

Detailed System Description

DFA states will include all the course numbers for the computer science major such as CMPT440, CMPT220, MATH130, etc. The application will act as a planning tool for students in need of a validator that identify the courses that they still need to take in order to be accepted into a course that they desire to enroll in. User interaction will be determined by the course

that they are trying to take as different courses have different prerequisites. A student that wants to take higher level courses will have more prerequisite requirements than those that are trying to enroll in lower level courses. Thus, this application will be more beneficial to students that are enrolling in high level courses as the prerequisites become more complicated. Below is the DFA diagram for the project:



Requirements

The only physical requirement for the application is a keyboard as Prerequisite Validator only requires user input.

Literature Survey

Applications and systems that are similar to Prerequisite Validator are Degreeworks and Schedule Planner. However, these systems do not inform the student whether or not they meet the requirements for the course they want to enroll in. Instead, they only provide information such as the prerequisite courses that are required for the course with no relation to the courses that the student has taken so far. Prerequisite Validator seeks to directly inform the user about the requirements based on what the user input/provided for the application. This means that the user can plan out their registration by only providing the course symbol of the courses they've already taken.

User Manual

Course Symbols:

```
BUS100 = a   MATH130 = b  MATH205 = c  MATH241 = d  CMPT120 = e  CMPT230 = f
CMPT330 = g  CMPT306 = h  CMPT307 = i  CMPT220 = j  CMPT308 = k  CMPT446 = l
CMPT422 = m  CMPT440 = n  CMPT424 = o  CMPT221 = p  CMPT331 = q  CMPT412 = r
CMPT414 = s  CMPT415 = t  CMPT333 = u  CMPT435 = v  CMPT432 = w  CMPT404 = x
```

Users will be prompted to enter the course symbols that they wish to take (this becomes the accepting state). Course information will be displayed for the user such as the name of the course and credits. The user will then be asked to enter the courses that they've taken so far or the sequence of courses that they wish to take leading to the course they've

selected. If the user input nothing, they will be asked to input a new course symbol (new accepting state). If they do not meet the requirements for the course then they will be informed that there are still prerequisites that they have to take. On the other hand, if the student has met the requirements, they will be informed that they can enroll in the class.

Conclusion

Prerequisite Validator can be extremely useful for students that are having a hard time with course registration when it comes to figuring out the requirements for a course. Currently, the application is only useable for computer science majors so an expansion into other major/minor can make this application a complete tool for all students.