Data Visualization

Spring 2020

Dr. Zhu Ying

02/21/2020

**Team members:**

Shibo Zhou (Coordinator) [szhou14@student.gsu.edu](mailto:szhou14@student.gsu.edu)

Ruqian Cheng [rcheng5@student.gsu.edu](mailto:rcheng5@student.gsu.edu)

Chengyong Zhao [czhao9@student.gsu.edu](mailto:czhao9@student.gsu.edu)

GitHub: <https://github.com/ichwin/DataViz>

**Project name:**

The term “**four-year plan**” is a suggested table provided by the college department. It guides students to register for a constant load of courses to help the students finish the program (usually undergraduate) in time. It can also give the academic advisor a clue of what is the progress the student has been made during a consoling session.

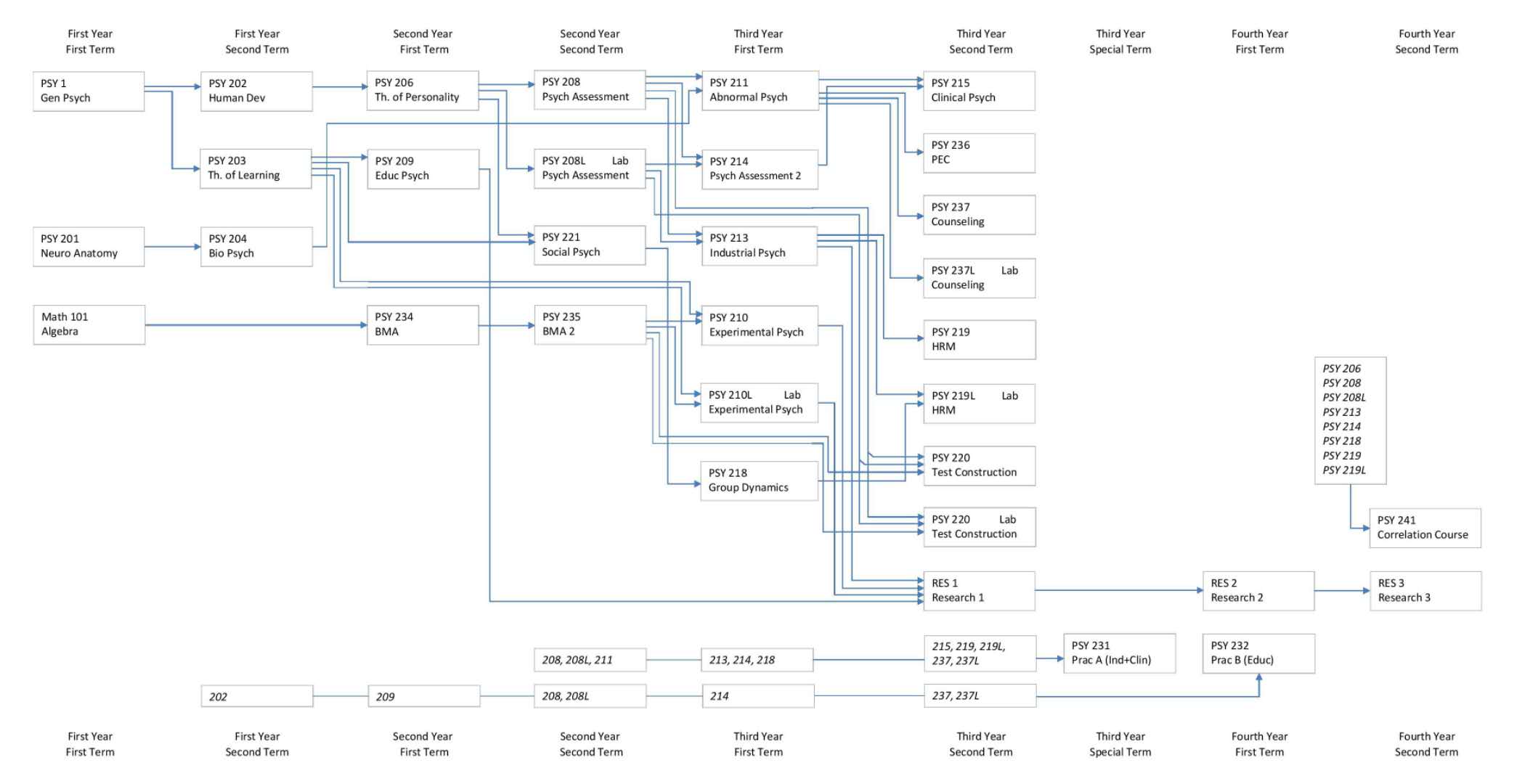
Some typical features of a “four-year plan” include:

1. All the program required core courses listed.
2. All their prerequisite course listed.
3. Ensured that for all semesters (at least for the first three academic years), there will be one free-elective course per semester, aiming to not make the general hardship too severe and balance.
4. Ensured that the cumulative course difficulty each semester grows gradually, leaving the students enough room to make the high school to college transition smoothly.

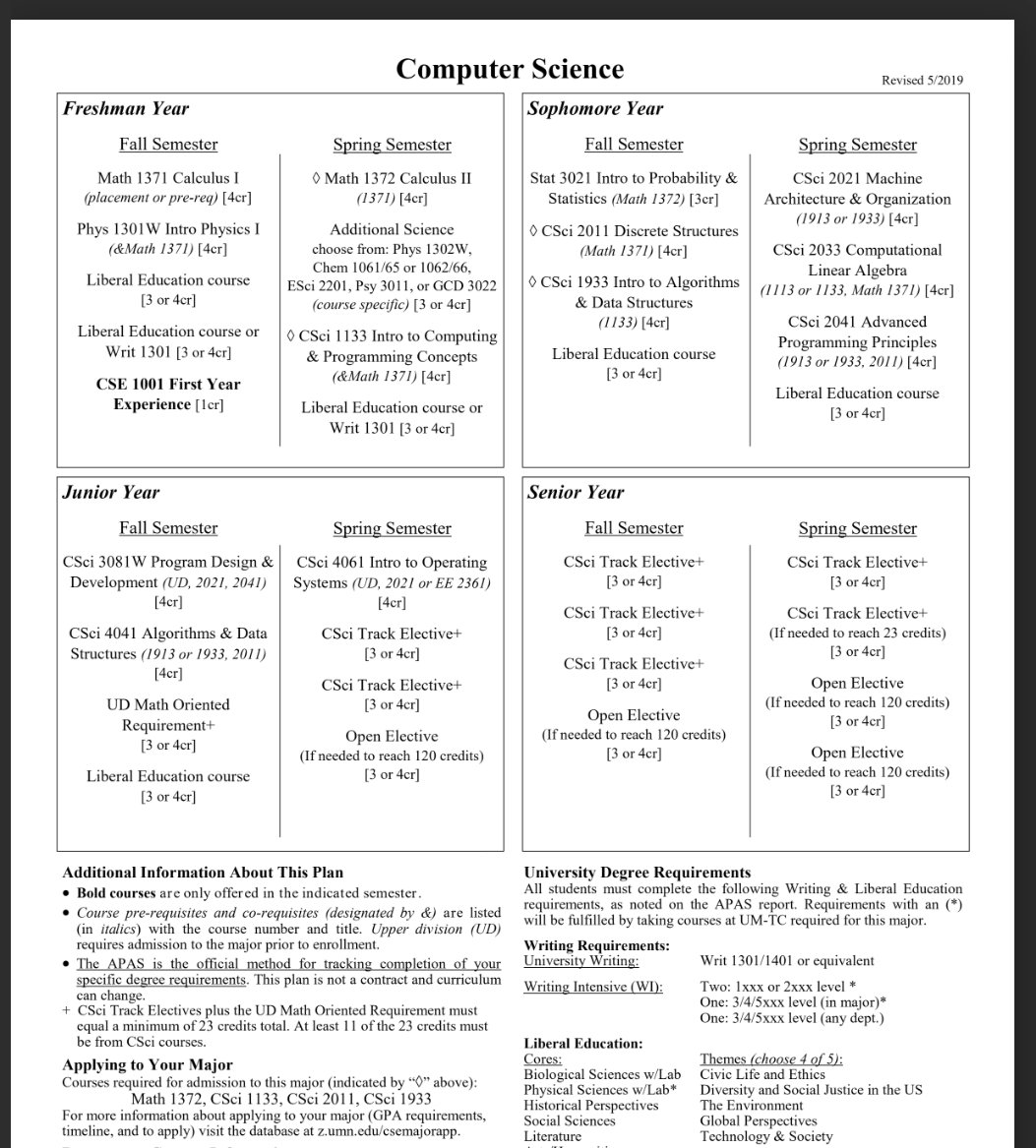
The plan may not be always followed. The course registration has always been competitive, and people who have registered for the desired course could drop it for various reasons, rendering the plan not being followed. Our project here is to make a dynamic data visualization of the four-year plan. Upon user preference, the four-year plan will have to be customizable and adjustable.

Good examples:

1. General Psychology undergraduate program of UCSD:



1. Computer Science undergraduate program of UMN:



Features that our “four-year plan” application will possess:

1. Import the existing course listed from the academic catalog in case any course was canceled. (supposedly work similarly like web crawler or parser)
2. Filter out the unavailable courses based on what courses are offered the current semester.

If two or more courses have a time conflict, visualize it.

1. Can adjust the distribution of cumulative difficulty by semester accordingly based on “strategy” to earn grades. For example, the maximum difficulty (increased core/elective course ratio) for easier senior year, or vise versa; graduate school applying strategy: minimum difficulty for best grade possible except for the last semester (by when the graduate school has already received the application); GPA rescuing strategy: minimum difficulty regardless of plan…
2. Can adjust the course load by semester accordingly based on “strategy” to complete the program. For example, maximum course load to quicken graduation; apply for grad school strategy: increased priority to graduate in summer, avoid graduation in Spring…
3. It can employ certain special rules defined by the user. Such as: “blacklisting” certain course; listing certain courses as “must”, manually increase the preference level of certain course to prepare for a potential major transferring…

The data visualization will be implemented using python as backend, Flask library for a web-based front end, GraphViz for data visualization after suggested by the instructor.