**Student requirements per course**

Created by John Hayes, Nick Demanche, and Evan Roberts

**General documentation for use of this program**

**Program Functions**

**DegreePath** - This definition holds all the code for the degree path functionality.

cisDisplay/cisPrint, cloudServiceDisplay/clsPrint, cyberSecurityDisplay/cysPrint - Each one contains the code for displaying the code from the PLP Degree path module.

**CoursePrediction** - Contains all the code for the course prediction functionality.

studentFile/courseFile - Both hold the code that opens the windows select file window.

dataComparison – This section holds all the code used in the actual comparison and the code to breakdown the data into its raw form.

**StartScreen** - This contains the code for the buttons and such for the main screen. And the main call and loop is located beneath it.

**Each comment from here down will describe the code labeled by the corresponding comment.**

#This part skims through the student data to grab only the tech courses.

The code here is eliminating the unnecessary courses and leaving behind the tech courses and removes the duplicates.

#This part checks if the student is passing the courses left behind and removes the failed courses, it then converts the student ID lines into strings of asterisks.

This code section removes courses that have failing grades. It also converts the student id lines into a string of asterisks and the courses are broken down into more basic components.

#This section cleans up the data and leaves behind only the essentials ex: CIS210 CYBD220 etc.

The code here cleans up the student data and prepares it for the comparison process.

#This part cleans up the course data csv file for later use.

This removes the parentheses M's and commas from the course file.

#This part converts the course list into a dictionary that allows the courses to have storable values.

The student course data here loaded a few lines previously is then converted to a dictionary and then a clone of it for the comparison.

#This is the data comparison part; it spits out a print of the dictionary with numbers relating to the number of students that need to take that course.

Does what the comment says it does.

#This gives the final value of the students.

The window prints the dictionary on the window and creates a txt file with the results as well.

#Comparison screen window code.

This contains the code for the buttons and window data.