1. File Purposes
    - courses.json
        - This file contains a dictionary of key and value of three key and value pairs which includes, the course name, course section, and credits. This file is used for the course registration of students.
    - student_registration.py
        - This is our main program. This file contains the code that initializes the class attributes for our “Student” and “Course” classes. The program opens the JSON file to utilize the functionality of adding/removing classes, waitlisting classes, and viewing a student schedule.

2. How to run the program
    - On the command line write “python3 student_registration.py courses.json” to run the program

3. How to interpret the program
    - Firstly you will be given the ability to add your student information which includes:
        - The students name
        - Student ID: Must start with “117” and be exactly 9 digits long
        - Student major: Must be one of the majors included in the “UMD_MAJORS” global variable dictionary
        - Student email: Must use a “@terpmail.umd.edu” email address
    - You will then have your student information displayed and be prompted with a question regarding what service you want to use
        - When adding or removing a course make sure to enter the course information as seen in the “courses.json” file
        - After you complete your desired service you will then be given the option to either add or remove another course or view your student schedule list(which you should be able to see after you start building your schedule by adding a course)
        - If the course is full then you can be added to a waitlist
        - Finally, after you are done using the interface you can select “4” to quit the program
4. Attribution

Method/function	Primary author	Techniques demonstrated
Student class __init__	Samarth	super(), regular expression
main()	Samarth	json.load(), fstrings
add_student	Chinedu	sequence unpacking, comprehension

remove_student	Chinedu	sequence unpacking, comprehension
Course class__init__	Dylan	With statement
get_student_enrollments	Samarth Motwani	Conditional expression
parse_args	Dylan	ArgumentParser class