Algorithms Specialist Tasks

The data structure specialist needs to first load the data into python and then clean the data (the first two bullet points under “Data Structures Specialist Tasks”)

* Develop linear and binary search algorithms to find specific student records or performance data.
  + Create two functions for searching for specific student records or performance data. The function should take in a parameter (a student record/name for example, or some performance criteria) and return the record(s).
    - 1. A linear search function: Loop through all the records (the data that the data structures specialist loaded into python and cleaned), and return the record(s) once found.
    - 2. A binary search function: Sort the data (use the built-in sort in numpy or pandas, or use on of the sorting algorithms created in the next task below). Next, perform a binary search on the data and return the record(s) when found.
* Implement sorting algorithms like selection, bubble, and insertion sort to order the dataset by various criteria.
  + Implement a number of sorting algorithms yourself (selection, bubble, insertion, I don’t know how many we need so maybe just these three are ok).
    - Create one function for each sorting algorithm. These functions will take the loaded and cleaned data from earlier and sort it and return the sorted data.
  + It says “by various criteria” so also add a parameter to the function that states what criteria to sort on (student name, age, etc).
* Compare the efficiency of custom search and sort algorithm implementations with those provided by Python's standard library or the optional libraries (Pandas/NumPy
  + Compare the various sorting functions that we created to the built-in sorting functions that numpy and pandas have.
  + The best way to compare is to time how long it takes for the functions to run and how long it takes for the numpy and pandas functions to run.
    - Can be done like this: <https://stackoverflow.com/questions/12444004/how-long-does-my-python-application-take-to-run>
* Use algorithmic techniques to find correlations or other significant relationships between different attributes in the dataset.
  + - Examples can be finding which genders have better grades, or which schools have better performing students of a certain age and gender.
* Apply the implemented algorithms to analyze trends and patterns within the student performance data.
  + Sort the data based on something such as gender or age and then find the average grades for each gender or each age. To find the average (or whatever else you choose), use the code that the data structures specialist will write as their 4th task.