**Project 3-B | Address (Contact) Book**

You will need to design, implement and test a “simple address book” application that manages a list of contacts (e.g., names, addresses & phone numbers).

**Academic Honesty Policy Reminder** – **Do your own work** – each submitted project will be compared against other submissions from current and previous semesters.

**Requirements**

* By using your application, users must be able to -
* **Add a new entry** (name, email addresses, mobile number and one additional piece of information of your choice) to their address book
  + **Required fields**: name + mobile number (a new entry should NOT be created unless the required information are provided)
  + Input validations – invalid phone numbers (non 10-digit number nor email addresses are not allowed)
* **Update (any portion of) an existing entry**
* **Delete an entry**
* **Search for an entry** (by name or phone number)
  + even a portion of a name; do you know how to spell **Arnold Schwarzenegger**?
* **Display current list of contacts**
* **Save a copy of the (latest) address book to a file** (prior to exiting the program)
* On start, load the most current address book
  + Create a new address book if one does not exist. (It will be the case when your address book is being used for the very first time)
* **We are working with the latest information at all time**
* Include your name as a comment in your code

**Examples**

* See **sample** test runs

**Start working on each project immediately so that we can discuss any concerns or questions you have.**

**Write-up & Submission Requirements**

**Review the provided rubric to understand the project expectations**, including documentation, CMSC206 and programming requirements. Each project submission MUST include your working Python code and a write-up.

Your write-up, at a minimum, needs to address -

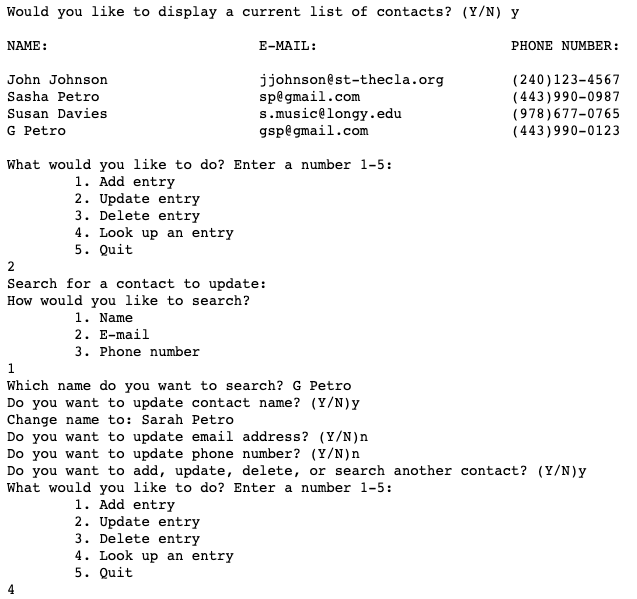
* **Approach, design & algorithm**
  + **DO NOT** start coding your project immediately! Come up with a high level design first. Complete this step first, then write your code
    - Break the project into smallest modules where applicable
  + Each student is welcome to expand on the design. If your project includes “above and beyond” features, clearly articulate them in your write-up. Potential extra credit could be earned if that is the case
* **Test plan & test cases**
  + What test cases did you run?
  + I want to understanding your “thinking,” and you need to be sure that your project works!
  + **For each test case, specify the expected vs. actual outputs**
  + Capture several test runs as screenshots, as you run your test cases
  + I will attempt to break your code (with reasonable test cases)
* **Highlight your learning experience and lessons learned**
  + I am very interested to learn about what you have done, how you did, etc.
* **~~Discuss your additional enhancement~~**
  + ~~What did you do?~~
* Any assumptions that you are making for this project
* Anything else that you want to share

Each student must submit one compressed (.zip) file back to the Assignment (link) with the following:

* Source code (.py or .ipynb files)
* **Write-up** (a Word or PDF file)
* Name your compressed file as <lastname>\_project\_x
  + where x is the project number and your last name (e.g., Thai\_Project\_1.zip)
* Review provided instructions on how to submit the project carefully (don’t assume anything)
* **I can only grade what’s being submitted. Double check your submission**
* **I MUST BE able to compile, run and test every submitted project on my computer using a set of test cases.** Just as important, I will READ your write-up first

**Not clear? That’s okay, but do ask your questions. “I did not know” or “I did not understand” is not good enough.**

**Sample Test Run – Your requirements have changed**

****