CS401 Project

1. Project Description

CS401 spring project is to build a coupon inventory system (CIS) using JAVA. This project exercises data structure topics discussed under this course. The project must use at least one user defined class and a main program class.

The system has following data field:

1. Name of Coupon Provider (max field: **20** bytes, e.g. Groupon, Living Social, DealDaddie, etc.)
2. Name of product (max field: **20** characters)
3. Price of product (e.g. **$10, $20, $50**, etc.)
4. Discount rate of the coupon (e.g. range between **5% and 80%**)
5. Expiration Period (e.g. range between **0 and 365** days)
6. Status of a coupon: Unused or Redeemed

- Let the project create the list by **both** a file input and by a manual user input. (FREE DESIGN) – means data input can come from reading a data file AND/OR from manual user input.

Create an unsorted list (UL) with **a linked list** implementation. This UL is your master list of this project.

User's initial input creates an UL.
When a user selects a menu, then construct a SL (Sorted List, depends on with a key field).
** Using one of these sorting algorithms: Quick Sort, Merge Sort or Heap Sort (you need to implement your own way – do not use a built in Java library class).

The UL also needs two different ways to create an array:

1. Default value of N (e.g. use N = **30**)
2. User input array size

Functions

1. Purchase Coupon (Input data)
   a. Coupon purchase (insert it into data collection) : either using data file and/or manual way (data file is provided by a user)
   b. Once importing data from a user provided file is completed, then a user continues to input more data manually

2. Search coupon(s) – user can search a coupon (or coupons) of a product
   a. If coupon is found, show coupon to a user
   b. Also provide count of search from the existing list **← provides number of searching to conclude**
   c. When searching, use both BST algorithm and linear search algorithm.
   d. For example, if there are 10 products in the list and the search finds the coupon in 4^{th} comparison by BST and 8^{th} iterations by linear search, then the count output: "**Found in 4^{th} by BST and 8^{th} by Linear Search**"
   e. If searching element is not found, then return "**No Coupon is found – 5^{th} by BST and 10^{th} by Linear Search**" **← provides number of searching to conclude**

3. List coupons – Provide ascending order of Per coupon site, name of product, price, discount rate, and expiration period, or Unused / Redeemed coupons

a. For example, if a user wants to see list of final price from LOW to HIGH, list products per final calculated price
b. The final price should be the price with discount rate
c. For example, if a price of a product is $20 and the discount rate is 20%, then the final price is $16

- No standard interface is given (it's your **free design**)
- If more functions are implemented well, extra credit points can be earned (up to 5%)

GUI is not required but if your project runs as full GUI frame and if your GUI is easy to operate, you may be qualified to earn extra credit (up to 5%). Command line window GUI is not considered as a full GUI. Since Java supports both styles of GUI, you may do either applet version or application version if GUI is implemented.

2. Project Requirements

- Your project has a menu to select each feature. (either command line menu or GUI)
- User interface: **FREE DESIGN** (expect no same interface among all projects)

Programming language: JAVA ONLY

What to submit?

Submit softcopy through the Course BB

1. Documents based on Software Development Life Cycle (in one file except source code files and a data file)
   a. Project managing schedule – daily progress plan
      Hours per each task to be done, etc.
   b. Problem specification – What problems are solving?
   c. Software specification - What functions are there?
   d. Design diagram document (including UML diagram)
   e. Operational document (user's manual: how to run your project)
   f. Testing document (your own created one) with input data file
   g. Debugging note (if available)
   h. Future improvement document (if available)
   i. Visual presentation screen shots (step by step) how your project works.
   j. A data file (or more files)
2. Source code files (*.java) – must include detail comments
3. Java bytecode files (*.class) – for command prompt/terminal run

- SDLC File name should include your Last and First name:
  - e.g. **CS401_proj_SDLC_Lastname_Firstname**.{doc/ppt/pdf}
  - Do not compress files. You can submit multiple files by one submission.
  - Above black color font part is common for everyone
  - Above blue color font part is your last name, first name order
- All documents except source codes formats: either PDF, PPT or MS Word

Final Project Due Date: ***April 24th, 2022 Sunday*** 23:59 Chicago time or earlier

- Late submission penalty: 10% in every hour (increase up to 60%)

  - e.g: submit on 4/29 1:05 AM – 20% penalty
  - e.g: submit on 4/29 6:01 AM or later - 40% of remaining project credit until noon
  - No acceptance from 4/25 Noon
- Demonstration to TA: Upon TA schedule announcement, you must be available.
  - Everyone must demonstrate his/her project to your TA. Your TA will ask QnA of your project and coding.

- This project is an **individual project**. Do not share your design or implementation with anyone. If similar part is found by your TA, those people (regardless of donor) will get failed this course. Look IIT student handbook for Academic Honesty Policy.