2XB3_2015 Assignment 3

Department of Computing and Software McMaster University
Instructor: Dr. Reza Samavi

February 27, 2015

This is an individual assignment. All assignments deemed to be substantially similar to each other will get 0 credit.

1. Description

McDonalds, Burger King and Wendy's are the biggest fast food restaurants in the world. All have many franchised locations at convenient corners in town. These restaurants compete on many levels from food selections to prices of meals. For example, they all offer a bundled meal (combo) which includes a sandwich (or burger), a soft drink, and a snack (usually fries). Recently, on top of its regular menu McDonalds launched a new campaign to promote one special half-price meal per day, 7 different meals a week. Its rivals, Burger King and Wendy's, quickly followed the suit. Later on, a one-item daily special is also launched due to the increase in competition.

Back to your home base, the university is closed for the reading week. After completing piles of assignments and several important exams, you and your friend are planning to drive to New York City. In order to get the most value out of the trip, you two are thinking of spending less on food so you are targeting either McDonalds, Burger King or Wendy's when you stop for a bite. However, there are many options to choose from and you do not want to eat two of the same sandwich consecutively. How will you quickly choose the best meal at the desired location along your trip? Well, there is still time before packing for the trip, so you decide to write an application to help your search.

2. Setup

In this assignment, there are two sets of menus for each restaurant: a list of regular meals and a list of special meals (item) of the day. The meal lists are posted on the course website named (menu.xlsx) with different tabs for "RegularMcDonalds", "SpecialMcDonalds", "RegularBurgerKing", "SpecialBurgerKing", "RegularWendys", and "SpecialWendys". The locations of restaurants are described by latitude and longitude in three files: "mcdonalds.csv", "burgerking.csv" and "wendys.csv". You and your friend are going to pick the meals according to the following requirements:

- The application should consider two peoples' choices (for you and your friend). You can only go to one restaurant at a time.
- Neither you nor your friend cannot repeat your prior meal choice (fries, snacks and soft drinks can be repeated though)
- Hardcode your day of departure into your application. You need this information as the restaurants offer different special menus for each day of the week. There are three stops on the way and the entire journey takes place in one day.

3. Assignment 3 Problems

3.1 Binary Search Tree (BST)

You are asked to implement a BST for the following problem. Please refer to section 3.2 in the textbook for the definition of a BST.

Problem: Your first stop is in Rochester, NY (77.6114^{o}W, 43.1656^{o}N). Within 50km, you want to have a beef sandwich or burger, with fries and a cola, and your friend wants a chicken sandwich or burger, with ginger ale and fries, at the best value. Return the results stating the restaurant(s), the address(es), the meal choices for you and your friend and the total cost of meals according to the format shown in Section 4..

3.2 Self-balancing Binary Search Tree (Self-balancing BST)

You are asked to implement Self-balancing BST for the following problem. Please refer to section 3.3 in the textbook for the definition of a Self-balancing BST.

Problem: Your second stop is in Newark, NJ (74.17257^{o}W, 40.72422^{o}N). Within 100km, you and your friend both want to have a meal with beef, cola and fries plus a milk shake, at the best value. However, this time you need to consider the fuel cost one way (aerial distance between two points on the map). Given the gas price is 63 cents/Litre, and your car's efficiency is at 8.2L/100km, return the restaurant(s), the address(es), the meal choices for you and your friend and the total cost of meals and fuel according to the format shown in Section 4.

3.3 Hash Table

You are asked to implement Hash Table for the following problem. Please refer to section 3.4 in the textbook for the definition of a Hash Table.

Problem: Your last stop is The Big Apple, New York City (74.0059^{o}W, 40.7127^{o}N). After parking your car at the motel at night, it is the best time to tour Times Square (73.9857^{o}W, 40.7577^{o}N) and have a late supper there. You want a beef with a cola, but you cannot decide whether you should have one more beef sandwich/burger or the fries to make it a combo. Your decision will be based on the cost. Your friend wants chicken and a cola, plus either a fries or another beef burger. Within 5km, return the results stating the closest restaurant, its address, the meal choices for you and your friend and the total cost of meals.

4. Output Specifications

You should have 3 different output files with respect to each problem above, named "a3_out_p1.txt", "a3_out_p2.txt" and "a3_out_p3.txt". Each output file should contain only the results of each problem above, presented in five columns, delimited by a tab character. The heading of these five columns is: the restaurant, address, the meal for you, the meal for your friend, and the total cost. It should end with a carriage return. The list of options should be shown on ascending order based on

the total cost. When you present the meal choice, it might have several items – please separate the items using commas as in the following example:

Meal for you	Meal for your friend
Big Mac - Meal, Ice Cream	McChicken, French Fries (Medium), Apple Pie

Assignment 3 Due Date

Friday, March 15th 2015, at 23:59.

Assignment 3 Marking

Assignment 3 is worth 5% of the course marks. Your grade for this assignment will be determined based on the following rules:

- A submitted solution that does not compile or run gets 0 credit.
- A solution that runs but is partially correct gets partial credit (depending on the progress towards a full solution).
- Providing adequate, concise, and meaningful comments throughout your code is part of the solution grade (i.e., a piece of code that correctly solves a problem without (or with inadequate) comments will score less than a well-commented piece of code that does the same).
- Your implementation should not only be correct but also concise and efficient. Quality aspects of your implementation and programming style particularly preservation of encapsulation and modular programming will be evaluated (refer to pages 96-108 of your textbook).
- Not following the assignment instructions properly for the requested formatting will cost you marks. You may even get 0 credit if the improper formatting will prevent your program from running.
- Every hour after an assignment deadline 2% will be deducted from the assignment mark. After 48 hours the assignment will no longer be accepted and the student will get 0 credit.
- This assignment is individual work. The work you submit must be your own. Both copying assignments and allowing others to copy your assignment are strictly forbidden and will be treated as an academic offence. All assignments deemed to be substantially similar to each other will get 0 credit.
- If you include libraries from any sources other than your own or from the course material (course lecture notes and lab notes/instructions) you must acknowledge them and explicitly give proper credit with meaningful comments inside your code (when using methods from the external libraries). Properly cited external codes can only be included as Java libraries, i.e. you are not allowed to copy full or partial codes from other resources and include them inside your code. The included libraries should not be a substantial part of your assignment. Your work will be checked for plagiarism to account for this.

Assignment 3 Submission

Your submission will be an Eclipse project of your A3 implementation. The name of your Eclipse project should be 2xb3_A3_*lastname_initials*. You should include a txt file named 2xb3_A3_*lastname_initials*.txt resided in the "data" folder containing the following information (each item in a separate line):

- The course code (COMP SCI 2XB3 or SFWR ENG 2XB3)
- Your full name

- Your student number
- A dated statement that attests to "the fact that the work being submitted by you is your individual work."
- Any design decisions you feel need explanation or attention by the marker (extra methods etc.).

In order to submit your Eclipse project to the relevant lab assignment dropbox in Avenue, you first need to save everything and then export your project.

1. In Eclipse, right-click on the name of the project, select Export->General->Archive File.
2. Ensure that just your project has a check-mark beside it, and select a path and filename to export the project to. Ensure that your export project has a file extension of '.zip'. The name of the zip file should be 2xb3_A4_*lastname_initials*.zip.
   IMPORTANT: You MUST export the FULL Eclipse project. Submitting individual files (e.g. java/class files) will NOT be counted towards your submission. Click 'Finish' to export.
3. Verify the zip file by opening it and ensuring that it has the same folder structure as in Eclipse (it may have some extra files or folder such as 'bin', which is okay).
4. Go to Avenue and upload your zipped project to ' Assignment 3 Submission' Dropbox.

IMPORTANT: YOU CAN SUBMIT MULTIPLE TIMES, HOWEVER ONLY THE LAST SUBMITTED FILE WILL BE CONSIDERED FOR MARKING AND ANY PREVIOUS SUBMISSION WILL BE AUTOAMTICALLY REMOVED FROM THE COURSE WEBSITE - IT IS YOUR RESPONSIBILITY TO CHECK YOUR ZIP FILE BEFORE SUBMITTING