Data Structures and Programming

Individual Assignment 1

Searching, Reading, and Writing Files

The purpose of this assignment is to familiarize you with C++ programming, data structures, and searching algorithms. For this assignment, you will be processing a recipe dataset. The recipe dataset file is called “IndianFoodDatasetCSV.csv”. You will also be given a searching algorithm to used when conducting searches in the program.

**The Problem**:

A client (Ms. Angelo) wants you to build a searchable recipe application. The application’s purpose is to allow users to search for a particular food recipe based on some criteria. With this application, the user will have several ways to search for a particular recipe in the dataset. Below is a list of requirements that you gathered from Ms. Angelo about how the application should work.

1. Greeting Interface: The first thing the application should do is greet the user and ask for the user’s first name. Once the user has correctly inputted their first name; the application will then move to the Main User Interface.
2. Main User Interface: The main user interface will ask that the user choose one option from the following options:
   1. Search for a Recipe by Name
   2. Search for a Recipe by Ingredients
   3. Search for a Recipe by Preparation Time
   4. Search for a Recipe by Cook Time
   5. Search for a Recipe by Cuisine
   6. Exit
3. Search for a Recipe by Name: When this option is selected, the user will be asked to input the name of the recipe they want to find. Once this input is received, the dataset is searched. The search should be for any recipe names beginning with the user’s input.
   1. if any recipes matching the input are found, then for each recipe display
      1. Name
      2. Ingredients (Note: When displaying this, display each ingredient on a separate line. The ingredients are separated by a comma.)
      3. Prep time
      4. Cook Time
      5. Servings
      6. Diet
      7. Instructions
   2. If no recipes are found, display “Sorry [USERNAME], No Recipes found”. Replace [USERNAME] with the user’s name as gathered from the input on the Greeting Interface.
   3. Following ‘a’ or ‘b’, wait for the user to indicate they want to return to the Main User Interface
4. Search for a Recipe by Ingredients: When this option is selected, the user will be asked to input up to three ingredient names. Once this input is received, the dataset is searched. The search should be for any recipes with the ingredients provided by the user.
5. if any recipes matching the input are found, then for each recipe display
   * 1. Name
     2. Ingredients (Note: When displaying this, display each ingredient on a separate line. The ingredients are separated by a comma.)
     3. Prep time
     4. Cook Time
     5. Servings
     6. Diet
     7. Instructions
6. If no recipes are found, display “Sorry [USERNAME], No Recipes found”. Replace [USERNAME] with the user’s name as gathered from the input on the Greeting Interface.
7. Following ‘a’ or ‘b’, wait for the user to indicate they want to return to the Main User Interface
8. Search for a Recipe by Preparation Time: Search for a Recipe by Ingredients: When this option is selected, the user will be asked to input a preparation time. Once this input is received, the dataset is searched. The search should be for any recipes with the preparation time provided by the user.
   1. if any recipes matching the input are found, then for each recipe display
      1. Name
      2. Ingredients (Note: When displaying this, display each ingredient on a separate line. The ingredients are separated by a comma.)
      3. Prep time
      4. Cook Time
      5. Servings
      6. Diet
      7. Instructions
   2. If no recipes are found, display “Sorry [USERNAME], No Recipes found”. Replace [USERNAME] with the user’s name as gathered from the input on the Greeting Interface.
   3. Following ‘a’ or ‘b’, wait for the user to indicate they want to return to the Main User Interface
9. Search for a Recipe by Cook Time: When this option is selected, the user will be asked to input a cook time. Once this input is received, the dataset is searched. The search should be for any recipes with the cook time provided by the user.
   1. if any recipes matching the input are found, then for each recipe display
      1. Name
      2. Ingredients (Note: When displaying this, display each ingredient on a separate line. The ingredients are separated by a comma.)
      3. Prep time
      4. Cook Time
      5. Servings
      6. Diet
      7. Instructions
   2. If no recipes are found, display “Sorry [USERNAME], No Recipes found”. Replace [USERNAME] with the user’s name as gathered from the input on the Greeting Interface.
   3. Following ‘a’ or ‘b’, wait for the user to indicate they want to return to the Main User Interface
10. Search for a Recipe by Cuisine: When this option is selected, the user will be asked to input a Cuisine. Once this input is received, the dataset is searched. The search should be for any recipes with the Cuisine provided by the user.
    1. if any recipes matching the input are found, then for each recipe display
       1. Name
       2. Ingredients (Note: When displaying this, display each ingredient on a separate line. The ingredients are separated by a comma.)
       3. Prep time
       4. Cook Time
       5. Servings
       6. Diet
       7. Instructions
    2. If no recipes are found, display “Sorry [USERNAME], No Recipes found”. Replace [USERNAME] with the user’s name as gathered from the input on the Greeting Interface.
    3. Following ‘a’ or ‘b’, wait for the user to indicate they want to return to the Main User Interface
11. Recipes should be found regardless of the case (capitalization) the user uses for input.
12. Your application must use the provided dataset.
13. The application should only close when the user chooses the “Exit” option from the user interface described in 2.
14. All of your user defined entities e.g. variables, functions etc should end with your initials.
15. A Microsoft Word document estimating the time complexity of the original search algorithm and the time complexity of the program you wrote should be submitted with the application.
16. A Microsoft Word “Help Document” which has screenshots and instructions on how to use the application should also be submitted with the application.
17. Upload your C++ files and OTHER DOCUMENTS to the LMS for grading.