External Documentation

**Program Name**: SmartFridge

**Authors:** Marty Yung and Youta Wu, students of CPSC 1150

**Purpose**:

The purpose of this program is to allow a user to store food in a refrigerator and to keep track of the quantity of each item stored. The fridge will automatically convert between units of the same type such as litres to millilitres or grams to kilograms. If required, the items can be displayed in alphabetical order. Also, a recipe can be imputed and the fridge will recognize which ingredients to dispense and in what quantities. If an ingredient is missing or there isn’t enough, the fridge will print a shopping list instead.

**Required Packages**:

This program requires the java.util.Scanner package, the java.util.Arrays package, the java.io.File package,and the java.io.PrintWriter.

**Classes:**

The class called SmartFridge contains the following methods: main, printMenu, exportFridge, addItem, removeItem, importFile, printFridge. The class called Recipe contains the following methods: enterRecipeMenu, printRecipeMenu, addRecipe, deleteRecipe, printRecipe, getMissingItems, and importFile

**Input:**

**a)** When adding items to the fridge, the user will be prompted for input as follows:

*What would you like to put in?*

In response the user should type a food item and press enter. For example, **milk**

Then the fridge will prompt the user to enter a quantity for the item they just put in as follows:

*What unit are you storing this in?*

User should type a quantity and then press enter. For example, **2 L**

**b)** When removing items from the fridge, the user will be prompted for input as follows:

*What would you like to remove?*

In response the user should type a food item. For example, **tomatoes**

Then the fridge will ask the user how much of the item to remove.

*What unit are you removing the item from?*

In response the user should type in a unit such as **5**.

**c)** When wanting to import a file, the fridge will ask the following:

*Please enter the file name of a .txt file to import from?*

The user should enter a text file such as **roastchicken.txt** and press enter in response.

**d)** To add a recipe, the fridge will display the following prompt:

*Please type a filename (no extensions) for the recipe:*

The user should enter a file name such as **butterchicken.**

**e)** When wanting to make a recipe. The following will display.

*Please enter the file name of a recipe you want to make?*

The user should input a file name such as **chocolatechipcookies**.

**Output:**

**a)** The program will display a list of items and their quantities

Tomatoes, 2

Orange juice, 1.4 L

Cheddar cheese, 85 g

Salsa, 400 mL

**b)** The program can also display a recipe already stored inside and show what is missing

You are missing 100 g ofchicken

You are missing 1 pepper

You are missing 150 mL of stock

**Limitations:**

1. This program cannot convert between units of different types such as metric to imperial. Attempting to input an item or taking it out in a unit that the fridge does not support will result in the following message being printed.

*There was a problem with the unit you selected. The item in the fridge is using kilograms.*

1. The item name typed in by the user will matter because the program does not distinguish between different spellings of the same item. This means “orange” and “oranges” will be stored as separate items instead of one because the program does not recognize that they are actually the same item.
2. Recipes must also have units that the fridge can convert to. For example, a recipe may call for a cup of orange juice but if the fridge stores it as litres then it wont know how much to dispense and an error will result.

**Improvements:**

1. This Smart Fridge could potentially become even smarter. To do this we can program the fridge such that it keeps track of the best before dates of every product stored. The user will also have the option of printing a list of best before dates for all items stored inside. When the best before date passes, the fridge can print out a message to the user to tell them to check on the specified product. The fridge will then give the user the option of either throwing it out or to keep it.
2. Another improvement could be to allow the fridge to convert between different unit types such as pounds to kilograms or litres to gallons.
3. The program can be improved such that the fridge will recognize that “orange” and “oranges” are the same item so that they will be stored in the same array index rather than two different indexes.
4. To further improve the recipe option, the fridge can be programmed to store multiple recipes. The user can choose how many recipes they would like to use and the fridge will display them. If more than one recipe calls for the same ingredient, the fridge will tell the user whether there is enough to make all the recipes or just one.
5. To build on the previous point, the fridge could suggest other recipes to complement the chosen recipe. For example, if the recipe chosen is roast chicken, the fridge may suggest a Caesar salad recipe as a side dish for the chicken. If the user chooses to follow the suggestion, the fridge can dispense the necessary ingredients and if they are not available, a shopping list can be created.

**Bugs:**

**[Usage]** The contents and recipes that are imported into the program MUST be in the correct format for it to properly import. A sample file is provided in their respective folders.

**[Usage]** A shopping list is created despite there being enough ingredients in the fridge.

**Algorithms:** In a separate document