

Programming Project / C++

– Recipe Management System –

Student 1: Totorean Daniel-Andrei

Student 2: Vasii Vlad Emil

I. Task Description

Student 1 is responsible for designing and implementing the Recipe class:

- Define Recipe attributes and methods (name, ingredients, getters/setters)
- Ensure proper encapsulation and class interface

Student 2 is responsible for the RecipeManager and CLI:

- Implement loading/saving recipes from/to recipes.txt
- Provide add, edit, delete, and view commands in main.cpp
- Manage command-line argument parsing and file I/O

II. Data Structures Used by the Team

- Recipe:
 - String name
 - Vector ingredients
 - Int ingredient count
- RecipeManager:
 - Vector recipes
 - load(), save(), add(), edit(), remove(), display()

III. File Structure

- recipes.txt:
 - Each recipe block separated by a blank line
- Format of each block:

<RecipeName>:

<N> Ingredients

<ingredient1><amount>

<ingredient2><amount>

...

IV. Interacting with Executable

To run the application, use the generated recipe_app.exe with these commands:

- .\recipe_app.exe view [searchTerm]
- .\recipe_app.exe add <Name> <IngCount> <Ing1> ... <IngN>
- .\recipe_app.exe edit <Index> <NewName> <NewCount> <NewIng1> ...
- .\recipe_app.exe delete <Index>

Note:

- No keyboard input; all I/O via command-line arguments.
- At least two classes (Recipe and RecipeManager) demonstrate association.