# PROJECT
# MANUAL

## *DIET PROBLEM SOLVER*

The objective of this solver is to identify the most cost-effective and nutritious combination of foods that will fulfill all daily nutritional requirements. The combination of foods will be based upon the food options selected by the user.
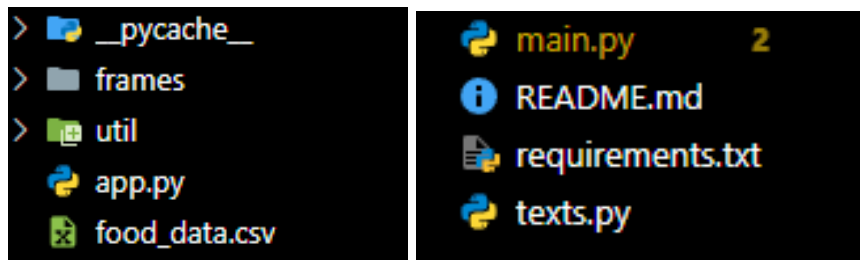
## COMPUTER SCIENCE

VICTORIA, Jan Clarisse B.

202208601 | AB1L

# Diet Problem Solver

The objective of this solver is to identify the most cost-effective and nutritious combination of foods that will fulfill all daily nutritional requirements. The combination of foods will be based upon the food options selected by the user. This problem is formulated as a linear program with the objective of minimizing cost under specified constraints and ensuring nutritional adequacy. These constraints control factors such as number of calories and amounts of vitamins, minerals, fats, sodium and cholesterol in the diet. Additionally, each food option is restricted to a range of 0-10 servings. The program employs a simplex method set up as a dual problem to solve for the optimal combination of foods.
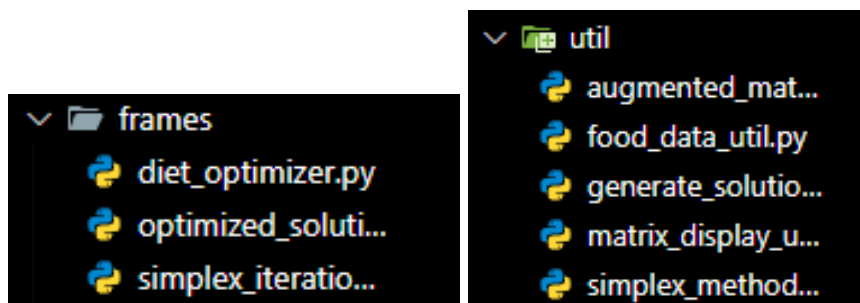
## Downloading Resources

Upon downloading all resources for the project, it should contain all the following files:

```
> __pycache__          main.py          2
> frames               README.md
> util                 requirements.txt
  app.py               texts.py
  food_data.csv
```

## Utils and Frames Folders

The utils folders contain all the logic and necessary functions for various methods and return the necessary outputs for the user to see through an interface. The frames folders on the other hand contain all the UI for the program. It contains different pages and also handles the error prompts and calls the corresponding util function needed.

```
∨ frames                    ∨ util
    diet_optimizer.py           augmented_mat...
    optimized_soluti...         food_data_util.py
    simplex_iteratio...         generate_solutio...
                                matrix_display_u...
                                simplex_method...
```

# Installing Dependencies

Make sure you have Python and pip installed on your system before running these commands.

```
numpy==2.2.0
pillow==11.0.0
tk==0.1.0
ttkbootstrap==1.10.1
```

```
pip install -r requirements.txt
```

# Running the Program

Once you are in the correct directory, run your Python script using the python command followed by the script's filename. If you are using Python 3, you might need to use python3 instead:

```
python main.py
```

# Main Menu

Upon running the main application, we are greeted with the app overview for the diet problem solver. By clicking the *START* button, we start the program.
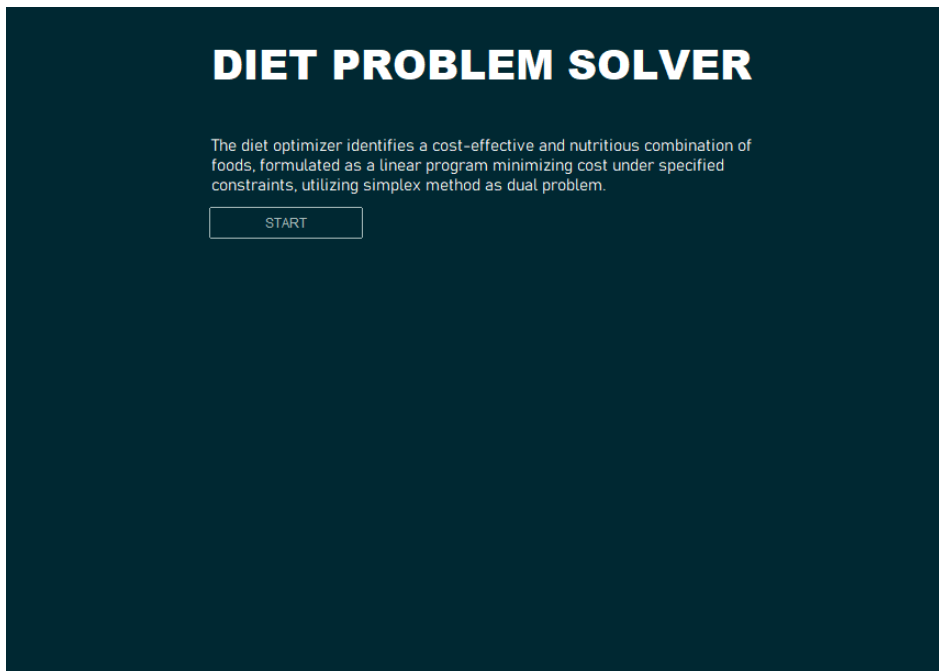
# Diet Optimizer

Now that the *START* button is clicked, the program will send the user to the diet optimizer where they have the option to select foods. The program has a built-in search function to make selection easier in addition to the default checkboxes.



Clicking the *SEE NUTRIENTS* button will show a pop-up window of all the food data with corresponding nutritional values and its price per serving.



The program also has a *SELECT ALL* function if the user chooses to have a wide variation of foods and for easier testing of the program's simplex method set up as a dual problem.

Upon clicking the *GENERATE* button the program will generate the optimal combination of foods that would meet the daily nutritional requirements at the cheapest cost. The *CLEAR SELECTION* enables clear all previous selections.



Upon clicking the *GENERATE* button without a food selected proper error prompts would be display and the *SEE SIMPLEX ITERATION* button would be disable as there is not food selection to optimize

However, upon choosing a variation of foods with a feasible solution, users can choose to see each simplex iteration with its corresponding basic solution. They can select the specific iteration they want to see and generate the iteration by clicking the *GENERATE ITERATION* BUTTON



This also includes proper error prompts if invalid iteration count was inputted by the user or if the user did not specify the iteration count upon clicking the *GENERATE* button