**Final Project Report: Meal Preparation**

GEC Course: Logical Thinking of Informatics

1. **Introduction**

In today’s hustle and bustle, finding time to plan a nutritious meal is so complicated. Why? Because going to the supermarket and figuring out which ingredients are needed for each meal is such a hard and time-consuming task. We have a bunch of different options for vegetables, fruits, meats, desserts and furthermore, we are not all experts in quantifying for a truly nutritious meal. For some people, it might be harder than writing a Python coding project. That is the reason why many people prefer choosing fast food, which doesn’t have a good impact on our health in the long term. No worries! We’ve got your back! Welcome to our Meal Preparation Program, we’re all about making healthy eating as easy as pie.

Are you also a member of the Planet Saver Team? Our new motto is "Balanced Bodies, Balanced Planet". We will present you with a selection of ingredients for your meal preparation based on your preferences and the various environmental implications. Our main goal for this project is to motivate people to have healthy eating habits and to protect our environment. By noticing small details in our daily life (a meal in this situation), we are one step closer to a healthy planet!

Let's get started; you'll find our project quite useful and worth sharing!

1. **Problems**
2. **Basal Metabolic Rate and Energy Requirement**

***Basal Metabolic Rate*** is the rate of energy expenditure per unit time by endothermic animals at rest[1](https://www.zotero.org/google-docs/?8xJc5H). In short, it represents the calories you need to stay alive. This includes basic functions such as:

* breathing
* heart rate and blood flow
* metabolism (digestion and nutrient absorption)
* cell function, growth and repair[2](https://www.zotero.org/google-docs/?mRNJI4)

Our program calculates your BMR for you. All you need to provide is your basic health information: age, height and weight. Using the Mifflin-St. Jeor equation, our program uses one of these two equations to obtain your BMR value[2](https://www.zotero.org/google-docs/?iU1lQl):

* Males: BMR = 10 × weight (in kilograms) + 6.25 × height (in centimeters) – 5 × age (in years) + 5
* Females: BMR = 10 × weight (in kilograms) + 6.25 × height (in centimeters) – 5 × age (in years) – 161

For example: A 22-year-old man who weighs 68 kg and is 175 cm tall will burn around 1668.75 kcal per day at rest.

Based on your BMR and physical activity level, our program determines your ***Energy Requirement*** for a day. We provide 5 levels of physical activity level with different value of *factor k*:

* sedentary (little to no exercise): k = 1.4
* light (light exercise 1 to 3 days per week): k = 1.6
* moderate (moderate exercise 6 to 7 days per week): k = 1.75
* intense (hard exercise every day): k = 1.9
* very intense (very hard exercise, training or physical job): k = 2.1

Your Daily Energy Requirement value is obtained by using the equation:

DER = k × BMR

For example: taking the same example of the young man above, if he has a moderate physical activity level then he needs 2920.31 kcal/day.

1. **Different combinations for a meal**

We believe that the most important thing for a meal is the variety of different nutrient sources and the balance of these different sources. That is the reason why in our program, we provide a meal composed of different nutrient factors:

* Carb Source, such as Rice, Potatoes, Maize
* Fat Source, such as Olive oil, Sunflower oil, Soybean oil
* Protein Source, such as Tofu, Eggs, Fish, Bovine Meat, Pig Meat, Cheese
* Vegetable, such as Tomatoes, Brassicas, Onions & Leeks
* Fruit, such as Bananas, Apples, Berries & Grapes
* Extra, such as Wine, Coffee, Dark Chocolate

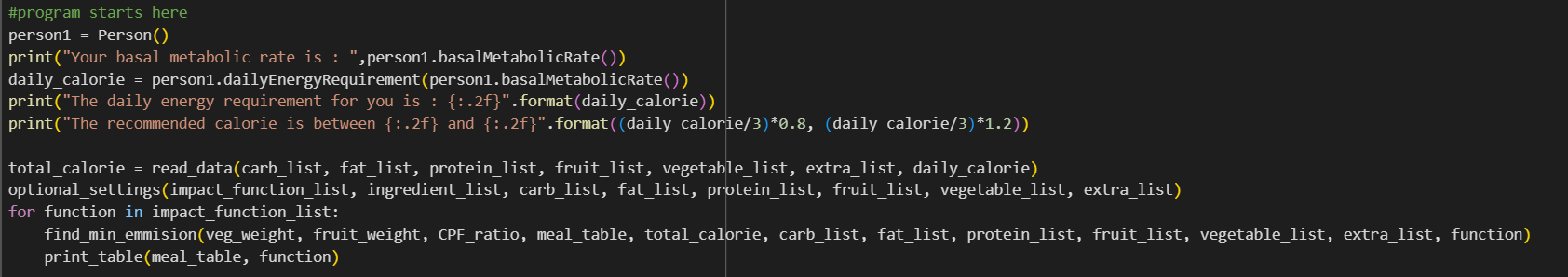
Furthermore, we are also aware that many of our users are concerned about environmental issues. We will go through our database to bring you the greatest diverse and nutritious meal composition with the least impact on our environment. We take into account 5 most important environmental impact indicators[3](https://www.zotero.org/google-docs/?aqMCQS):

* Land use
* GHG emission: greenhouse gases emission calculated by CO2eq
* Acidifying emissions: terrestrial acidification
* Eutrophying emissions
* Freshwater withdrawals

1. **Code Introduction**

Our code is composed of six code cells. Of these, only the sixth cell is the main function. The first cell mounts Google Drive and asks the user to upload the two files “Emission.csv” and “Nutrient.csv”, which can be found in “Materials”, the fifth section of this report. The second to fifth cells are class and function definitions.

We wrote our program with modularisation and object-orientation in mind to maximize its readability. All the object and function names intuitively demonstrate what they do. If we exclude variable declarations and focus only on lines with function calls, which is the part that comes after the comment “#program starts here”, the entire main function consists only of ten lines.



Let us take a look at these ten lines in detail. In the first line, the program constructs the “Person” class, and collects the user’s sex, weight, height and age. Then, the program puts the information collected into the basalMetabolicRate() method to compute the user’s BMR using the formula mentioned in the previous section. In the third line, the program uses the dailyEnergyRequirement() method to collect the user’s physical level, and computes the user’s daily energy requirement. The calculated result is then output by the fourth and fifth line.

The sixth line calls the read\_data() function, which reads and stores relevant data in the files “Emission.csv” and “Nutrient.csv”. The properties of each food are stored in “Ingredient” class objects, which are then appended to different lists based on their nutrient classification (e.g., carbohydrate, fat, protein…). The ingredients are put into different lists so that the program can easily find the least environmentally harmful combination by iterating through every possible combination. There are 67,500 combinations in total, which a human would find impossible to compute, but for a computer, this is a perfectly manageable number of permutations.

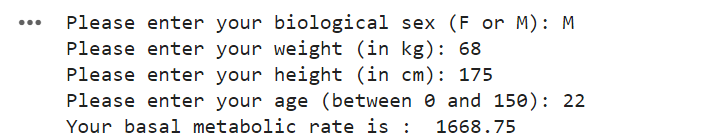
The seventh line reads in optional settings that the user may wish to include. These include specifying certain ingredients to include in the meal, and specifying the environmental impact the program tries to minimize. The specification of ingredients is, in fact, done by changing the previously mentioned nutrition lists. For example, the carb\_list originally contains 15 carbohydrate ingredients. If the user specifies “Rice”, then the carb\_list is changed to contain only “Rice”. This serves the additional purpose of reducing the amount of permutation needed.

The eighth to tenth line then computes the meal with minimum impact for every environmental impact the user specifies. As previously mentioned, this is done by permuting through every possible combination of ingredients.

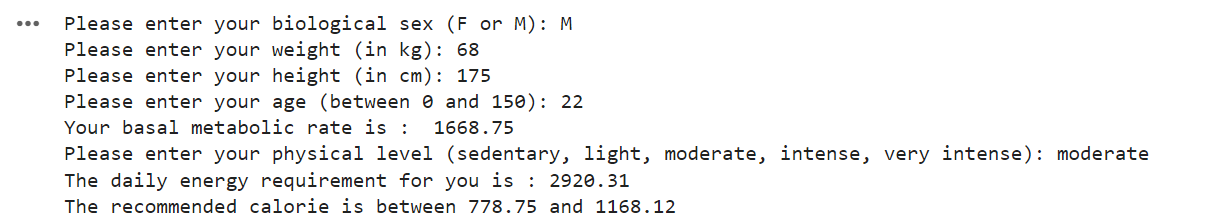
A final thing to note is that our program is designed to handle unexpected inputs. In the first five lines related to the “Person” class, these are achieved through the else case in if-else statements. In the last five lines, these are achieved through the try-except structure surrounding each input. A user can type virtually anything into the inputs that come after entering the physical level without crashing the program.

1. **Result**
2. **Personal information input and BMR, DER output**

Let’s consider the same example with the young 22-year-old man who weighs 68 kg and is 175 cm tall. This is the output for BMR and DER calculation of our program:

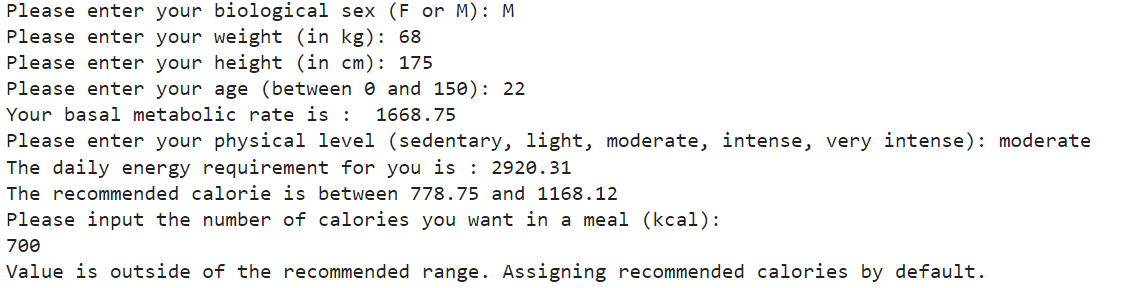


Supposing that he has a moderate physical activity level then his energy requirement is:

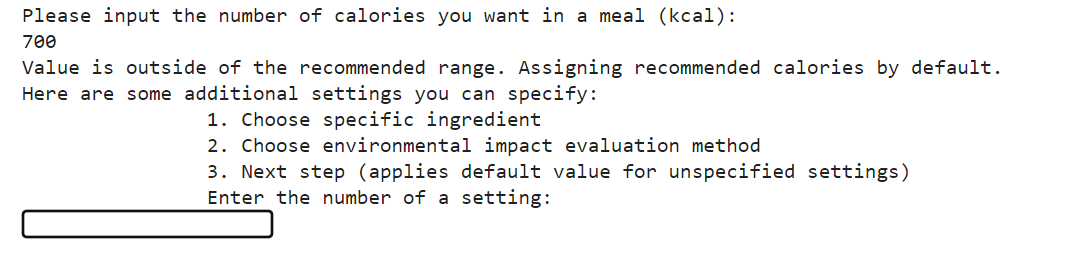


1. **Different features for a meal composition**
2. **Quantitative calories value for each meal**

After obtaining the value of his daily energy requirement, we supposed that the calories required for each meal is equal to ⅓ of this total value and set a range ±20% of this average value for each meal. To avoid harmful weight loss and lessen the overweight problem, this range is vital to maintain a healthy body with proper eating habits. If the user enters a value out of this range, we will assign a default value which is equal to ⅓ of his or her DER.

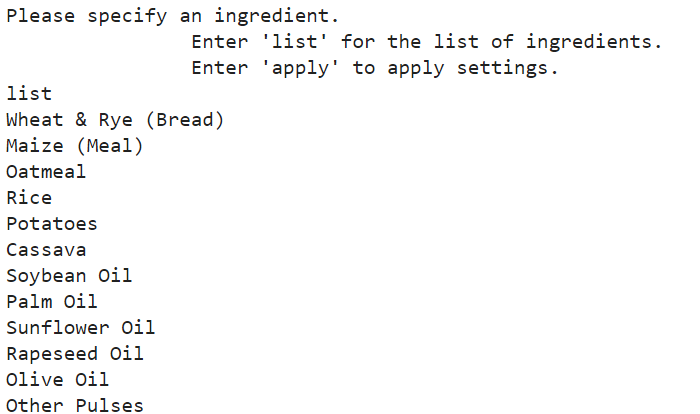


In the next steps, users can choose to specify certain ingredients existing in our provided list that they want to include in their meal or certain environmental indicators that they want to minimize.

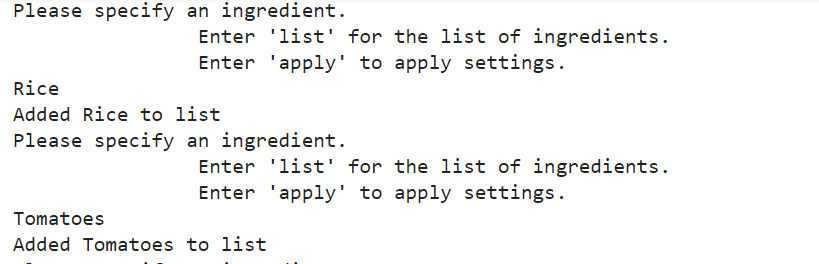


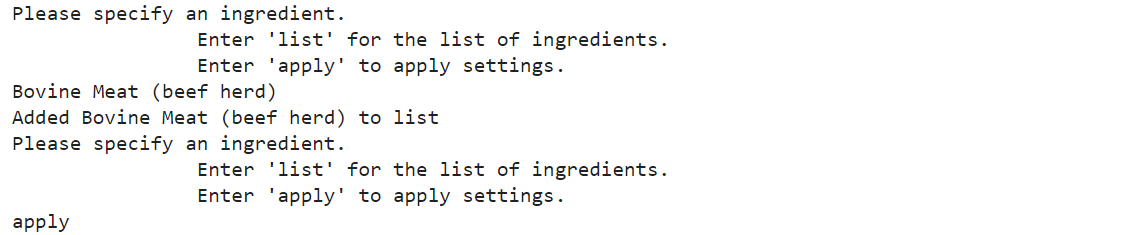
1. **Ingredients specification**

In this section, when the user chooses 1, they can view the entire list of the ingredients provided by our program by typing “list” in the input box. If they have no specific preferences, they can type “apply” in the input box and the system will automatically provide a nutritious meal based on its own calculation.

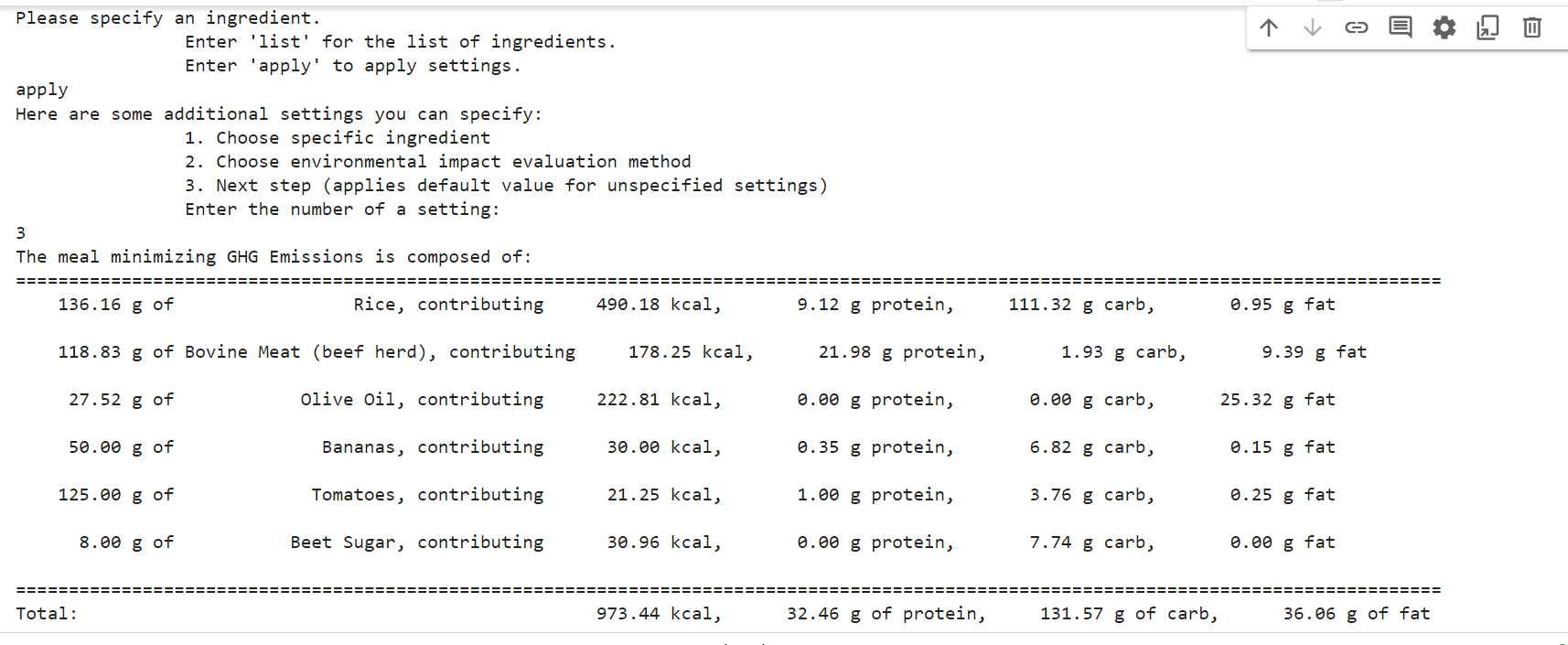


For example, let’s consider that the user would like to have Rice, Tomatoes and Bovine Meat for dinner. Please note that the user doesn’t need to choose a specific ingredient for each nutrient factor.





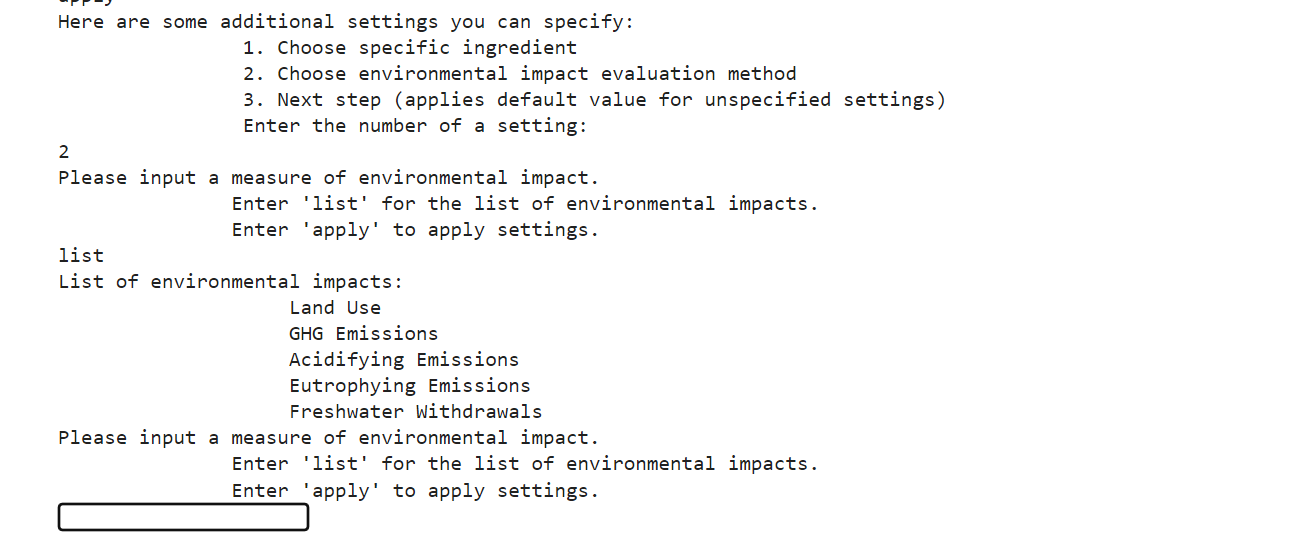
Once the choice is done, the user can type “apply” and move to the next step by typing 2 in the input box, or by typing 3 without specifying to minimize any environmental indicators. By default, the system would choose to take into account the indicator GHG emissions and would provide the “best” meal having the least environmental impact based on its calculation. For example, in this situation the user would type 3, let’s see what our program gives.



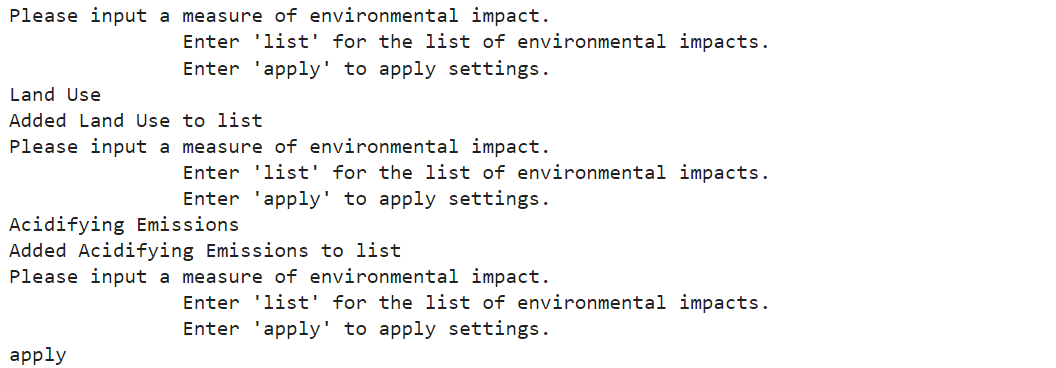
We can see that our program provides a table with detailed quantities for each nutrient factor and detailed calories for each of these ingredients.

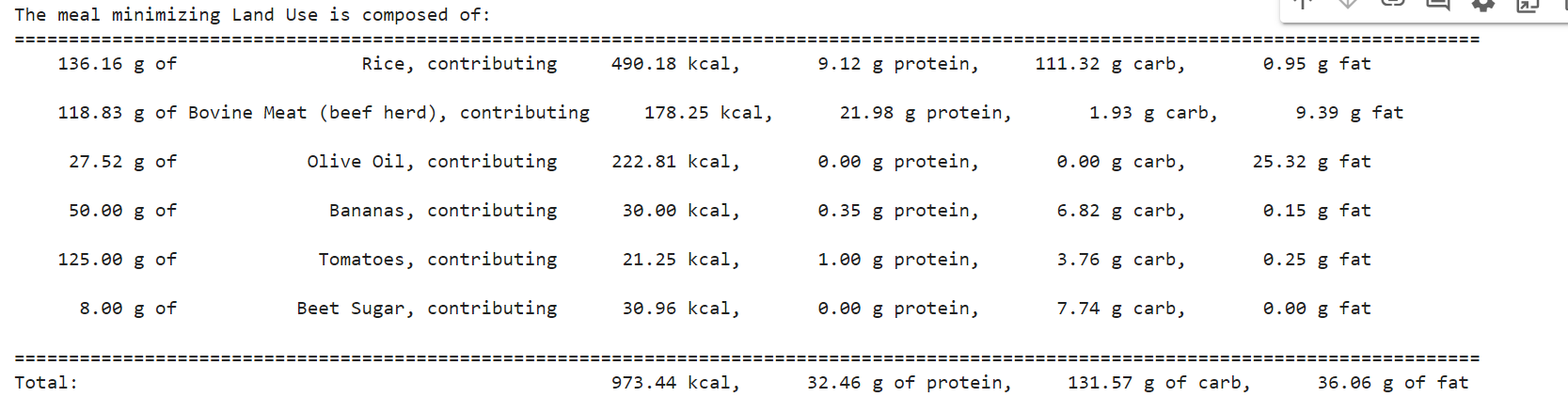
1. **Environmental indicators specification**

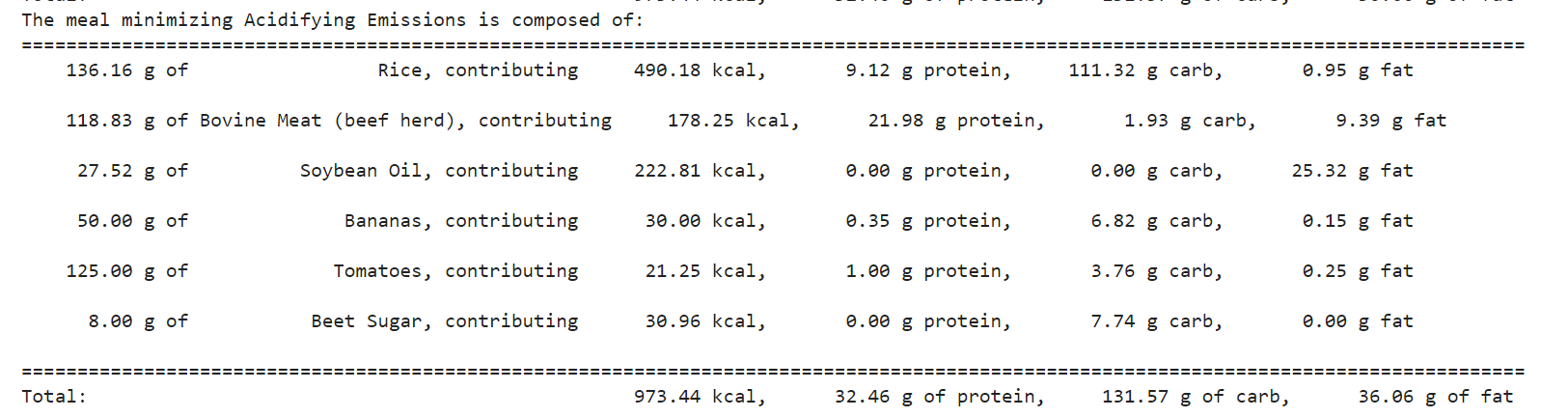
After choosing the specific ingredients in section 1, the user can continue to define their environmental indicators specification in section 2. Let’s suppose that the user chose 2 instead of 3 after taking Rice, Tomatoes and Bovine Meat into their list.



For example, the user would like to minimize the indicator Land Use and Acidifying Emissions. Let’s see what the system gives.



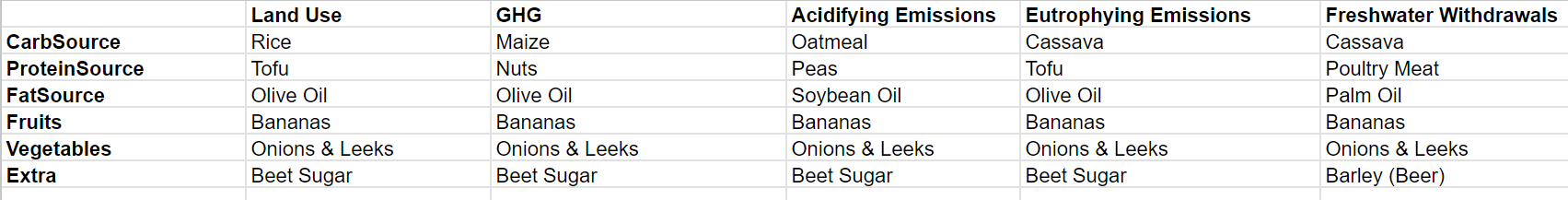




Our program would provide 2 different meals minimizing 2 different environmental indicators. However, by comparing 3 different meals with 3 different indicators, we can see that there is no big difference in the 3 combinations, which could be interpreted that by minimizing one environmental indicator, the others are not so high either. This comparison leads to an interesting remark that we found while playing with our program.

1. **Interesting result**

Firstly, we decided to leave section 1 empty by not indicating any specific ingredients. Then we try to minimize different environmental indicators to see different outputs of the program. Different combinations are represented in this image below.



We can see that Fruits and Vegetables are the same for 5 scenarios. Extra remains the same for 4 situations, except for Freshwater Withdrawals. Fat Source is similar for Land Use, GHG and Eutrophying Emissions. However, for Carb Source and Protein Source, the choice given by our program is relatively varied. We suggested that relations between these 5 environmental indicators might exist, there could be (negative or positive) correlation between different factors.

1. **Materials**
2. **Our dataset**

Our program requires two csv files to run: Nutrient.csv and Emission.csv. These files are submitted to eLearn along with our report and are attached below.

[Modified Dataset](https://drive.google.com/drive/folders/1w6GI4OTe6DjxV7WvhVNq_jELuuuRpUEO?usp=sharing)

1. **Our Colab notebook**

You can get access to our notebook via the link below. Please follow the instructions in the notebook to run the program.

<https://colab.research.google.com/drive/14L66t9U_uqEQWpZZcXl35gnXmozGcGPs?usp=sharing>

1. **Our video**

You can find our video presenting this project via the link below.

[Group20\_GEC1506\_FinalProjectVideo.mp4](https://drive.google.com/file/d/1BVAxNfZSqKFI9ZU0Ar2SP5hc6K2fM56P/view?usp=drive_link)

1. **Group task distribution**

* Chu Phuong Thao (X1120083) :
* Codes in the following cells:
* Person class and associated functions to calculate the basal metabolic rate
* Main function
* Writing the report
* Li Wei-an (109022119) :
* Writing the “Code Introduction” section of the report
* Codes in the following cells:
* Ingredient class and its assignment functions
* Functions that find and output the “best” meal
* Functions that read in additional users settings
* Main function
* Tieu Tuan Vi (110006129) : Video recording, editing and subtitling

1. **Bibliography**

[1. Basal metabolic rate. *Wikipedia* (2024).](https://www.zotero.org/google-docs/?OUAYKE)

[2. Basal Metabolic Rate Calculator. *Healthline* https://www.healthline.com/health/how-to-calculate-your-basal-metabolic-rate (2022).](https://www.zotero.org/google-docs/?OUAYKE)

[3. Poore, J. & Nemecek, T. Reducing food’s environmental impacts through producers and consumers. *Science* **360**, 987–992 (2018).](https://www.zotero.org/google-docs/?OUAYKE)