Question 15.2

In the videos, we saw the "diet problem". (The diet problem is one of the first large-scale optimization problems to be studied in practice. Back in the 1930's and 40's, the Army wanted to meet the nutritional requirements of its soldiers while minimizing the cost.) In this homework you get to solve a diet problem with real data. The data is given in the file `diet.xls`.

1. Formulate an optimization model (a linear program) to find the cheapest diet that satisfies the maximum and minimum daily nutrition constraints, and solve it using PuLP. Turn in your code and the solution. (The optimal solution should be a diet of air-popped popcorn, poached eggs, oranges, raw iceberg lettuce, raw celery, and frozen broccoli. UGH!)
2. Please add to your model the following constraints (which might require adding more variables) and solve the new model:
   a. If a food is selected, then a minimum of 1/10 serving must be chosen. (Hint: now you will need two variables for each food *i*: whether it is chosen, and how much is part of the diet. You'll also need to write a constraint to link them.)
   b. Many people dislike celery and frozen broccoli. So at most one, but not both, can be selected.
   c. To get day-to-day variety in protein, at least 3 kinds of meat/poultry/fish/eggs must be selected. [If something is ambiguous (e.g., should bean-and-bacon soup be considered meat?), just call it whatever you think is appropriate – I want you to learn how to write this type of constraint, but I don't really care whether we agree on how to classify foods!]

If you want to see what a more full-sized problem would look like, try solving your models for the file `diet_large.xlsx`, which is a low-cholesterol diet model (rather than minimizing cost, the goal is to minimize cholesterol intake). I don't know anyone who'd want to eat this diet – the optimal solution includes dried chrysanthemum garland, raw beluga whale flipper, freeze-dried parsley, etc. – which shows why it's necessary to add additional constraints beyond the basic ones we saw in the video!

[**Note**: there are many optimal solutions, all with zero cholesterol, so you might get a different one. It probably won't be much more appetizing than mine.]

NOTE: PuLP is a free solver that you can use with Python wherever you happen to be. However, PuLP isn't the best; the best solver these days – the one that on average solves models the fastest and can solve the largest models – is called Gurobi. Gurobi has a set of Python libraries called gurobipy that can be called from Python code just like any other library.

If Gurobi is so good, why not teach it instead of PuLP? Because, as a state-of-the-art commercial software package, Gurobi isn't universally free, and I want you to have a tool you can use without having to pay for a license. However, Gurobi *is* free for students in its full, unlimited form. Otherwise, if you're not in academia, the free version of Gurobi is limited to 2000 variables and 2000 constraints, which means (for example) it can solve the model for diet.xls but not for diet_large.xlsx.

Since you are a student, you can get a full Gurobi license and use gurobipy to solve both models, and I'll provide a set of .py files showing those solutions too. If you want to learn more about how to use Gurobi and gurobipy, their website has some nice examples and documentation. And, if you want to learn more about how to build and solve optimization models with gurobipy, you can check out my free four-part course "Optimization Through the Lens of Data Science" on Udemy.