Programming Assignments #4 and 5
CS 202 Programming Systems

For This Program

With both programs #4 and #5 you will be implementing your solutions using Java. Your goal must be to develop an object-oriented solution but this time implement it in Java. Make sure that your OO Design is not centered around your data structures – your data structures support the design but shouldn't be the primary emphasis of your design. You may use Eclipse or IntelliJ to develop your software. Your Java programs must follow these rules:

- **No public or friendly fields (data members) *** NONE!!!*****
- No friendly methods (member functions); this means the public, protected, or private keyword should be placed in front of each method.
- Limit your use of static methods – these should be restricted to just utility functions and main
- With your OO design, develop an inheritance hierarchy using "extends"; there must be a minimum of <u>5 classes with 3 of them in a hierarchy</u>. *These should <u>not</u> be isolated to just your data structures.*
- The application that USES the hierarchy must be in a class or hierarchy of its own.
- Constructors:
  - Implement at least one constructor with arguments
  - Use the super keyword in invoking a base class' constructor. *This is what we use instead of an initialization list. ***Write about this in your*
- Support dynamic binding with this programming assignment
  - In your OO design, look for the self-similar behavior between classes – that is where dynamic binding applies
  - In your design write-up discuss the difference between the functions that are being overridden versus those being overloaded
  - Have at least one abstract base class
- Implement at least two functions using function overloading between classes and experiment with the way function overloading works in Java.
- And yes, you SHOULD use the string class!

For each of the above that you experiment with, write up information about it in your efficiency write-up

You are required to turn in a paper on how this solution is object oriented (your design). There is ONLY ONE design writeup for the combined programs 4-5. But, with EACH program, you are required to write 400 words about the efficiency and IDE (rather than the debugger).

Program Requirements
Do you love pizza? I do! It is great but still painful to order. Half of the time when I order by phone (or even in person) they get it wrong. Or, when I login and use an online program – I have to filter through so much just to get to the pizza I want (I suppose I need to save my settings!!). A great app that would be easy to use for ordering pizza is much needed! And, wouldn't it be great if it was independent of the actual pizzerias? I'd like to order pizza the same way regardless of what pizzeria makes the pizza itself. Then the app contact the store and get it ordered for me!

Your job is to create a program in two phases. Think of it as something that would ultimately be an App to assist people in creating and ordering a pizza that they want delivered. Before the program can be used, we need to set up how a user will put together the desired toppings for the pizza that they want to eat. This will be the first step with Program #4. Then, once the data exists, with Program #5 we will have the program find the best match with the closest pizzeria.

Design Specifics: If you have never ordered pizza online, start by going to Domino's website or something similar. Get an idea of the different categories of topics and pizza. There are choices for the type of crust, sauces, cheese, protein (e.g., meat), vegies, and other possible toppings. Most places also have special combo pizzas already created and a special of the day. Think about what information the person ordering might want to know and what makes one type of toppings different from another – such as how many calories, the nutritional value or price. In your design push up the common elements for the different kinds of toppings. Think about how dynamic binding can be used to create a unique pizza with a set of desired toppings.

Another part of the ordering process should be combo pizzas already "specified" where the user can add more toppings or remove toppings. Each time the user logs in, a new special combo pizza should be advertised – randomly created.

Your job is to come up with a design of an OO framework. Although you all have the ability to write this using procedural abstraction, the key is to make sure to solve this problem using Object Oriented methodologies **with dynamic binding** and function overloading. *The use of external data file(s) is necessary!* ***Don't HARD CODE the toping specifics*** – *as that would not be real-world! For example, to add Kale as a new topping, it wouldn't make sense to have to hire a programmer to update the software!*

Data Structures
In these last two programs, you must implement two data structures:
1. Program 4: A doubly linked list of arrays
    a. The doubly linked list is the list of pizzas ordered so that you can easily retrieve the user's last pizza order.
    b. Use the array to store a specific pizza's information. Each pizza could have a crust, sauce, protein(optional), vegies (optional), other toppings. With dynamic binding, this can be an array of base class references.

2. Program 5: A tree of trees (e.g., *each node in the outer tree has a root pointer to another tree*); a suggested use would be for a tree of pizzerias and for each the pizzas available (and prices) that that establishment
    a. If you have already implemented a balanced tree in a previous assignment: A Binary Search tree, implementing insert, display, retrieve, retrieve all related items, remove an individual item, and remove all; the algorithms must be implemented recursively.
    b. If you have not yet implemented a balanced tree, then:
    A balanced tree, implementing insert, display, retrieve, retrieve all related items and remove_all (no remove individual items for a balanced tree).

The required data structures specified in the assignment must be your own implementation: as in BSTs (or balanced tree) and doubly linked lists. Once you meet the basic requirements of the assignment, you are allowed to use libraries for any subsequent data structures.