Jerome Veix
4/28/2022
Programming Language Concepts

Pseudocode for the Individual Project

```
1. Import all of the necessary packages
2. class Bank (TASK COMPLETED: SUPERCLASS)
        Set the data members that will be used in the accessor methods (TASK COMPLETED: DATA MEMBERS)

        Retrieve a variable that was previously created with the get() function
            return variable
        Repeat this process for as many data members as we have (TASK COMPLETED: ACCESSOR METHODS)

        Set the updated versions of the data members
            variable = newer variable name
        Repeat this process for as many data members as we have (TASK COMPLETED: MUTATOR METHODS)

        Create a mathematical operation using a data member
            {return the result}
        Repeat process with the same operation but now with an added element (FIRST STEP OF METHOD OVERLOAD)

        Use void method print("Sorry to inform you but, the bank is closed today.")
        (FIRST STEP OF METHOD OVERRIDE)

        Declaration of static method (FIRST STEP OF STATIC METHOD AND FIRST STEP OF POLYMORPHISM)

        Create data member
        Set the beginning value of the member
        Create method name
        Establish how program will repeat
        Establish the limit of time program will repeat
            print("Maybe this class isn't too bad")
        Call method (FIRST STEP OF RECURSION)

        Define what happens inside the show method (FIRST STEP OF AGGREGATION)

3. interface class name
        Set up a data member, must include public void
        Repeat another time (TASK COMPLETED: INTERFACE WITH TWO
        ABSTRACT METHODS)

4.  Create a class that is a subclass of the parent class that was produced beforehand
    (TASK COMPLETED: SUBCLASS)
        Set the data types that will be used in the accessor methods (TASK
        COMPLETED: DATA MEMBERS)

        Retrieve a variable that was previously created with the get() function
            return variable (TASK COMPLETED: ACCESSOR METHODS)

        Set the updated versions of the data members
            variable = newer variable name
        Repeat this process for as many data members as we have (TASK
        COMPLETED: MUTATOR METHODS)

        Use void method print("The bank is open today.")  (SECOND STEP OF
        METHOD OVERRIDE)

        Declare a static method
            print("The realtor name is John Smith.") (TASK COMPLETED: STATIC
        METHOD)
```

In the main function:

```
        Use scanner
        Create string variables

        Ask the user to input file PATH
        Ask the user to input output file name
        Enter the file name and enter the file name after
        close file

        Open reading connection, which is the Scanner
        Then open the output connection, which is the PrintWriter

        Read from the input file name
            Read characters from internal file
            Call the input string and convert to uppercase
            Print output

        Close the input and output connections (TASK COMPLETED: TEXT INPUT AND
        OUTPUT)
```

Look to alleviate the IOexception error if it comes up after running the code **(TASK COMPLETED: EXCEPTION CLASS)**

Create new object from the subclass
Calls the mutator methods to set the values of the variables
Use the get function to print results

Apply two separate calls of the method that differs between their number of arguments within parentheses **(TASK COMPLETED: METHOD OVERLOAD)**

Combine the new object with the old method to override the execution of the first instance of the method and instead runs the second one that is present in the subclass **(TASK COMPLETED: METHOD OVERRIDE)**

Call the declared static method **(TASK COMPLETED: STATIC METHOD AND POLYMORPHISM)**

Create empty stack
Add elements to stack
Print out the elements of the stack in the form of an array **(TASK COMPLETED: ARRAY BASED DATA STRUCTURE SELECTED FROM STACK)**

Method calls itself **(TASK COMPLETED: RECURSION)**

Create an ArrayList
Add integers to the list
Use Collections.binarySearch to identify an int within list
print variable that held the binary search **(TASK COMPLETED: SEARCHING ALGORITHM USING BINARY SEARCH)**

Create a new object from the superclass
Combine the object and show method **(TASK COMPLETED: AGGREGATION)**