**CS 1450 – Intro to Object Oriented Programming**

**Final Assignment (250 points)**

Course instructor: Patrik Boloz

Deadline: 5/12/2023 by 11PM (uploaded to Brightspace)

Your task for this final assignment is to produce computer code in the Python language and upload all required/generated files to Brightspace by the deadline. Please read the instructions carefully and follow them to get the full number of points. To achieve full points, each numbered assignment is worth 50 points, therefore completing all totals 250 points. You will upload one file per each numbered assignment; therefore, you will upload 5 .py files. Any code uploaded that does not successfully run will be considered incomplete and will result in only partial point assignment, which will depend solely at my discretion. The use of AI tools (like ChatGPT) and straight up copying of code from the internet will be considered as cheating, giving automatic 0 points for this assignment.

The numbered assignments are as follows:

1) Write a program that uses the for loop to find the factorial of a user given number between 0 and 50. If a user defined number is negative or bigger than 50, print out a message saying that the calculation cannot be performed. Print out the result in the terminal.

Hint: Use if else conditions. The factorial symbol: ! means to multiply all whole numbers from the chosen number down to 1. For example, if a user inputs 5, then the following calculation will happen: 5! = 5 \* 4 \* 3 \* 2 \* 1 = 120.

2) Given a list of numbers. write a program to turn every item of a list into its square.

Hint: For example, if a list is: [1,2,3,4,5,6], then the output list will look like: [1,4,9,16,25,36]. Use a for loop to iterate through your list.

3) Create a function called weather\_report(name) that will take one string parameter called name and will greet that name. Also create a list of different string weather situations and print out the current weather.

Hint: Use the random library and the function random.choice to choose a weather randomly from your list.

4) Create a Hotel class, that will have one parameter in the init method called name, so whenever the hotel object is created, you specify the name of the hotel. This hotel will also start with an empty guest list. Create three methods:

1) add\_guest – this method will add a specific name to the guest list and will print out a message that the specific guest has checked in.

2) check\_out\_guest – this method will remove a specific name from the guest list and will print out a message that the specific guest has checked out.

3) get\_guest\_list – this method will return the guest list

Then test out your new class by creating an object, add 4 guests, print out the list, check out the guest, and print out the final guest list.

5) Create a parent class called Building and 2 child classes that inherit from the parent class Building called Storage and Corner\_Store. The Building parent class will have 2 init parameters: name (name of the building) and size. Size will have a default value of “Medium”. The init attributes will be self.name and self.hours (list of 2 integer values). You will create these following methods:

1) set\_open\_hours(self, open\_time, close\_time) - this method will set the attribute self.hours to the hours the store will open and close. It will then print out a message saying that the store’s opening hours have been changed and show the hours. Hint: Use the list index bracket notation to assign the different hours

2) set\_name(self, new\_name) - this method will update the building’s name to a new one and will print out a message saying that the store’s name has been updated and what the new name is.

3) get\_name, get\_size, get\_hours - these methods will return their respective attributes of self.name, self.size, and self.hours

Then by using inheritance create 2 child classes Storage and Corner\_Store, where the only differences are that Storage will have a default building size of “Big” and Corner\_Store “Small”.

Create one object per each class, use the set\_open\_hours method to update each object’s opening hours, add the objects to a buildings list, and print out each buildings name, opening hours, and size with the use of a for loop.

Hint: For inheritance, there is no need to use the super() function, just repeat the init method with the default values.