CMIS 2720 Data Structures and Algorithms for Games

Ying Zhu

Assignment #4

Due date: 11:59 pm, March 31, 2023

In this assignment, you will learn the following:

- How to implement a graph and solve a problem on it
- How to implement a simple game inventory system using a Dictionary<T> class (Hash Table)

General information

1. You must write your programs in C#.
2. This assignment contains two separate programs: A4a and A4b.
3. Submit a zip file to iCollege under the folder Assessments → Assignments → Assignment4. The zip file should contain two separate C# files:
    a. firstname_lastname_A4a.cs
    b. firstname_lastname_A4b.cs

A4a requirements

1. Implement an UndirectedGraph class. This class should contain the following properties and methods.
    a. Properties:
        i. Nodes: a list of nodes in the graph
        ii. Edges: a list of edges (links) in the graph
    b. Methods:
        i. RandomGraph(): Creates a random graph
        ii. AddEdge(int NodeID1, int NodeID2): Add a new edge connecting Node1 and Node2 in the graph
        iii. RemoveEdge(int NodeID1, int NodeID2): Remove the edge connecting Node1 and Node2 (if any)

2. Write a program that does the following:
    a. At the beginning of the program, automatically create a random graph with 20 nodes. Some of the nodes are connected, and some are not. The connections are randomly chosen by the program.
        i. Use the graph you have implemented for A3a.1 (see above).
        ii. <u>Do not ask the user to enter the nodes and links.</u>

    b. Your program should be able to process the following commands from the user:
        i. **Print**: Print the adjacency matrix of the graph

        1. Even if you don't use an adjacency matrix internally to implement the graph, your program still need to display an adjacency matrix.

    ii. **Path**: Query if a path exists between node1 and node2. The query can be like "path 3 9", which means "Is there a path between node3 and node9?" The program replies either Yes or No, depending on whether a path exists between the two nodes.
        1. The program does not need to find and print the path between node1 and node 2. Just reply Yes or No.
    iii. Exit

3. Write comments in your code. If there is no comment, I will deduct 5 points.

A4b requirements

In this assignment, you will simulate a simple game inventory system.

1. You must use a Hash Table in this part of the assignment. You can use the Dictionary<TKey,TValue> class in C#/.NET or implement your own.

2. There is a central inventory system that stores some game items. Each item has a name (string), a power value (integer), and a price (integer). The player needs to buy items from the inventory system for a quest.

    a. Use a Has Table to implement the central inventory system.
    b. Use another Hash Table to implement the player's person collection of items.
    c. The player has 100 coins. Therefore, the player's collection of items cannot exceed 100 coins.
    d. If the player bought item A from the inventory system, the player can also sell item A back to the inventory system.
    e. There can be identical items in the inventory system. For example, there can be 3 identical swords.

3. At the beginning of the game, your program should automatically create an inventory system and add up to 20 items to the inventory.
    a. You decide the items and their prices.
    b. Do this automatically. Do not ask the user to add the items to the inventory system.

4. Your program should be able to accept the following commands from users:
    f. Buy: Buy an item from the inventory and add it to the player's collection.

   g. Sell: Sell an item bought earlier back to the inventory.
      i. The player can only sell an item bought earlier from the inventory. The player cannot generate a new item.
   h. Inventory: Display a list of items in the inventory, including their power and price.
   i. Collection: Display a list of items the player has bought, including their power and prices.

5. You must use Hash Tables in your program. I will check the source code. If Hash Tables are not used, you will get 0 credit.

6. Write comments in your code. If there is no comment, I will deduct 5 points.