List 04 – Data Structures – Hash Table

1) Create a C# program that simulates a game inventory.

Class Inventory

Stores objects of type Item.

- Implemented using Hashtable (System.Collections).
- Each item occupies a number of slots (defined by the property slotsOccupation in the Item class).
- Has a maximum number of occupiable slots (defined when creating the inventory).

Methods

- bool insert(Item item): If there is free space (the number of slots occupied by the item) in the inventory, insert that item into the inventory using the item's name as the main key and mark its slots as occupied.
- Item remove(String name): if an item with the given name is found, remove it from the inventory and return it. If an item is removed, free the number of slots it occupied. If not found, return a null reference.
- int numberItems(): returns the number of items stored.
- int numberSlots(): returns the number of occupied slots.
- void printItems(): prints the names of the stored items.
- void clean(): deletes all items from the inventory.
- void save(String filename): saves the maximum number of slots and the inventory items to a file named filename.
- void load(String filename): loads the items from the file into the inventory using the maximum number of slots specified in the file.

Class Item

Defines game items.

Attributes

- String name;
- enum Type type; (weapon, armor, potion, scroll)
- enum Rarity rarity; (common, uncommon, rare, epic, legendary)
- int numberSlots;
- bool uniqueUse;

Items must be created using a constructor and must have the corresponding getters.