Muddy Potatoes
Clarence Cheng, Tania Cao, Qian Zhou

APCS1 pd1
Jan. 3, 2018

Final Project Proposal

Juan's Delivery Co.

OVERVIEW

This project will be a survival simulation game. The user will first will be given an income, hunger and thirst levels, and an inventory of items. They will then be able choose between moving to the next day, shopping for items, or selling items from their inventory:

Whenever they decide to move to the next day, their income will increase and their hunger and thirst levels will increase by a random amount. If the user has food items in their inventory, the foods will be consumed and their hunger/thirst levels are decreased. However, if the user doesn't have Food items or Water items in their inventory and their hunger corresponding levels reach 0, the user dies. In addition, the user's income may randomly decrease when a thief steals money from the user.

When shopping for items, the user will search for the store they want to shop in, and then select items to buy. In random cases, certain items may be on sale or may be pricier than normal based on availability and supply demand. When purchasing, the user can purchase from more than one store and get more than one item from each. When the user is finished purchasing, the user will be given options to choose the cost of the delivery service and delivery time. Once the selection is complete, the program will display the total cost and delivery time. However due to random unforeseen circumstances delivery time may increase or decrease.

When the user shops for un-consumable items, the user may sell them for profit. To sell, they will select the item(s) from their inventory based on the price offered at that moment. Once the items are sold, the money will be added to their income immediately.

At any point, the user can check their income balance, hunger and thirst levels, the delivery status of their undelivered items, and their inventory upon request. If the user is able to survive for 10 days, they win the the game.

PROGRAM OUTLINE

Each type of item would be a subclass of Class Item. Class Item would have subclasses indicating types of items as Water, Food, and UnConsumable Items each in turn would branch out to names of certain Items. Each item would be instances of the subclass of the item type. Items in general would have attributes/variables such as cost, ID, and delivery time. Water and Food would have values as to how much of the corresponding bar they replenish. SuperClass Shop would have subclasses of different shops with different items in sell, different delivery times and different prices.

User would be able to interact with the program by means of implementations of Keyboard.java in the directory cs1 from the library. ArrayList could be used as a means of stimulating the inventory. Private and public variables will certainly be used.

To-Do List:

1. Driver class (Woo.java), to be changed corresponding to development
2. Abstract class Item
3. Abstract classes Food, Water, NonConsumable
4. User Class
5. Shop Superclass
6. Classes of specific shops
7. Specific Item classes
8.

Minimum stuff to add:

Prioritized to-do list.

Rough timeline.

How you are making meaningful use of OOP. (Visuals!)

How you are making meaningful use of at least two (2) other first term concepts.

- InstanceOf?

Stretch?