MIDTERM EXAM

Name: Chuhan Zhou

Student Number: 002839760

- This is a**n open book open notes.**
- Exam is to be **submitted by the due time**.
    - Exam starts at 12:00 PM and is due by 2:30 PM
- Please **submit a single word file or a single PDF file.**
- The exam has 5 questions with **varying weights**.
- **Partial Credit will be awarded for all questions.**

Q1 Class Diagram	\30
Q2 State Diagram	\30
Q3 Use case Model	\10
Q4 Use Case Scenario	\10
Q5 Java Interfaces	\20
Total	**\100**

Good luck!

Part I: Application Design

You are creating an application for Walmart. The Application is designed to help Walmart understand customers behaviors and provide information to guide Walmart stocking and placing items on shelves.

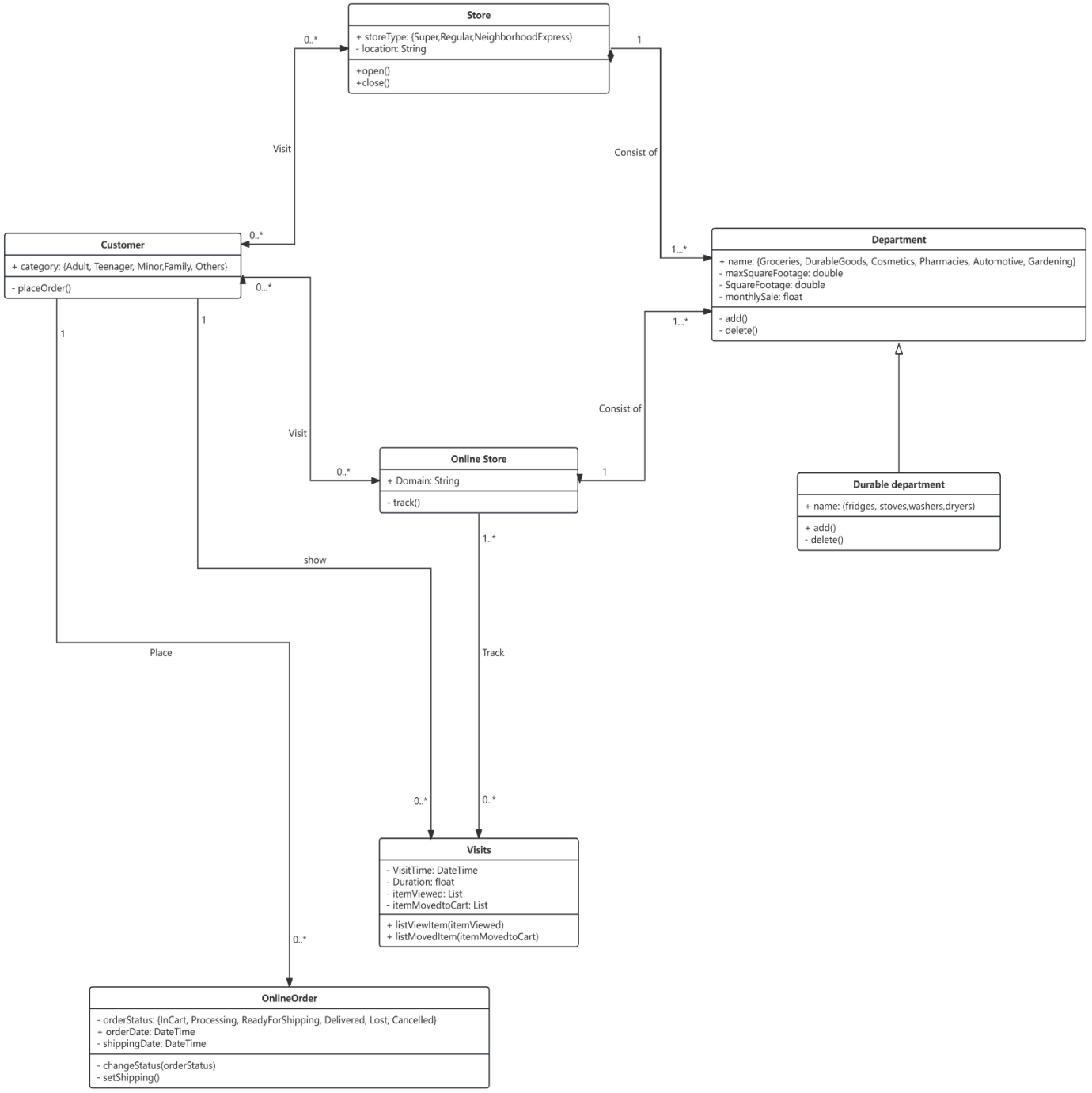
Walmart has a number of stores in Toronto. Stores are either super stores, regular stores, or neighborhood express stores. Each store is divided up into departments, including Groceries, Durable Goods (like fridges, stoves, washers and dryers), cosmetics, pharmacies, automotive, and gardening. Each department is allocated a maximum amount of square footage. But at any point of time, the amount of square footage allocated to a department may be less than the maximum threshold. Walmart allocate square footage to departments based on their monthly sales. If a certain department sales drops, its square footage goes under review and reevaluation.

Walmart tracks the customers who walk into their stores through the cameras. Walmart classifies customers into categories, such as adult, teenager, minor, family and others. Walmart tracks such information to help find the most suitable items for all customers.

Walmart also has online stores that sells similar items and has the same departments as discussed earlier. Online stores track visits and the browsing habits of its visitors (i.e. time of the visit, duration, items viewed, items moved to cart, etc). Walmart tries to keep similar prices for instore and online items. But occasionally, an item maybe available for less online.

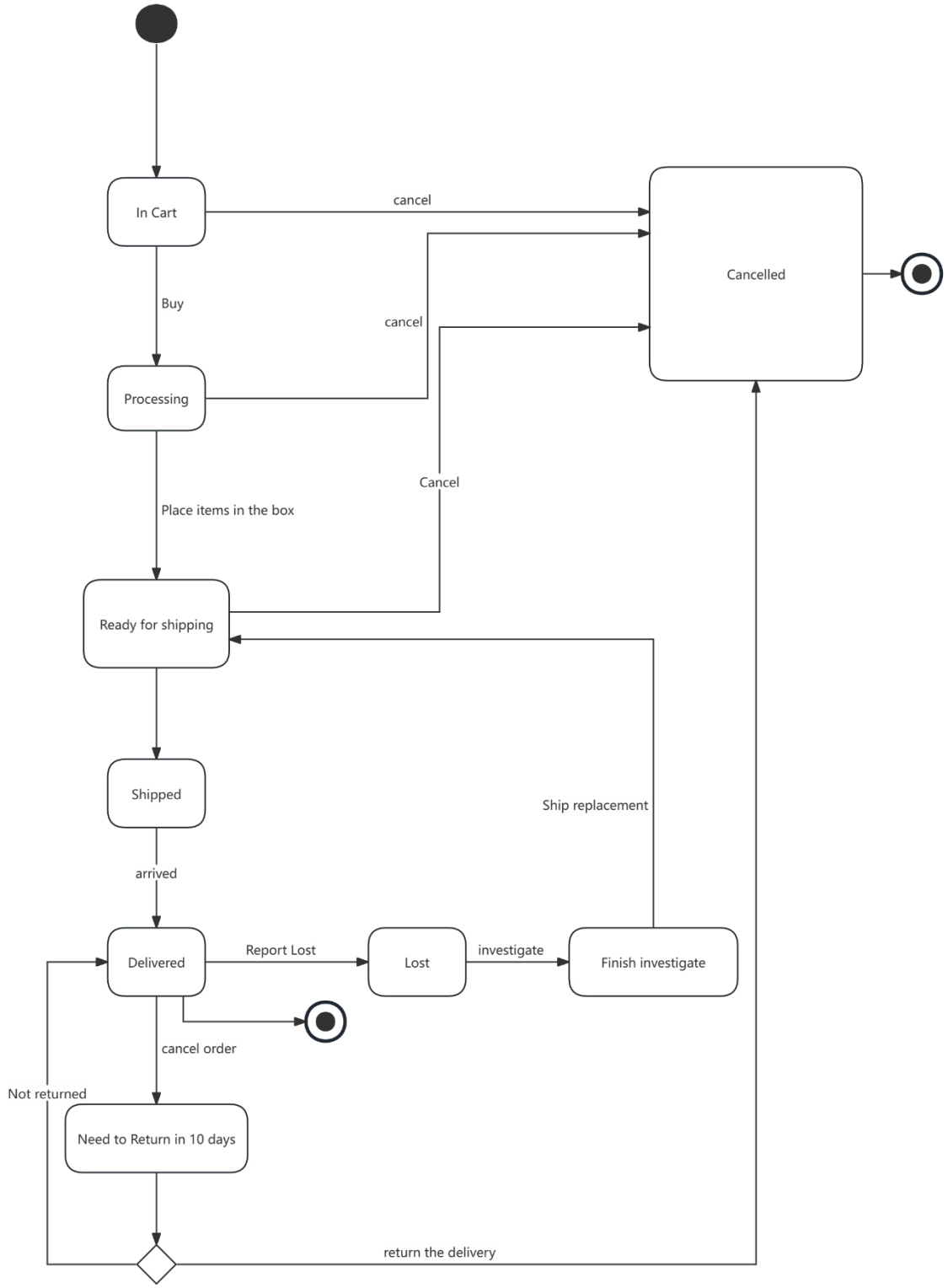
Walmart tracks online orders from the moment a customer places items in their cart. Once the customer clicks on check out, the order becomes ready for processing. Once the order is paid for by the client, the order becomes under processing. At that point, a Walmart associate must prepare the order. Once the items in the order and prepared and placed in the box, the order becomes ready for shipping. Another department takes on the order and gets it delivered to the client. Clients have 30 days to return an order. The order could be lost if the client reports it missing after the delivery. In this case, another department investigates the issue. A replacement as a new order must be shipped to the client within a week of it being reported lost. Clients can cancel an order at any time. If the order has already been shipped when it was cancelled, then the client must return the items to a Walmart store within 10 days of cancellation, otherwise the order will be treated as a regular order.

Q1. Create a **Class Diagram** for this application. Include at minimum all underlined items. Your class diagram should include all associations, multiplicities, and attributes. Add a few association names and roles names as needed. **[30 points]**


Store
+ storeType: {Super,Regular,NeighborhoodExpress}
- location: String
+open()
+close()
0..*
1
Visit
Consist of
Customer
+ category: {Adult, Teenager, Minor,Family, Others}
- placeOrder()
0..*
0...*
1...*
Department
+ name: {Groceries, DurableGoods, Cosmetics, Pharmacies, Automotive, Gardening}
- maxSquareFootage: double
- SquareFootage: double
- monthlySale: float
- add()
- delete()
1...*
1
1
Consist of
Visit
Online Store
+ Domain: String
- track()
0..*
1
Durable department
+ name: (fridges, stoves,washers,dryers)
+ add()
- delete()
1..*
show
Place
Track
0..*
0..*
Visits
- VisitTime: DateTime
- Duration: float
- itemViewed: List
- itemMovedtoCart: List
+ listViewItem(itemViewed)
+ listMovedItem(itemMovedtoCart)
0..*
OnlineOrder
- orderStatus: {InCart, Processing, ReadyForShipping, Delivered, Lost, Cancelled}
+ orderDate: DateTime
- shippingDate: DateTime
- changeStatus(orderStatus)
- setShipping()

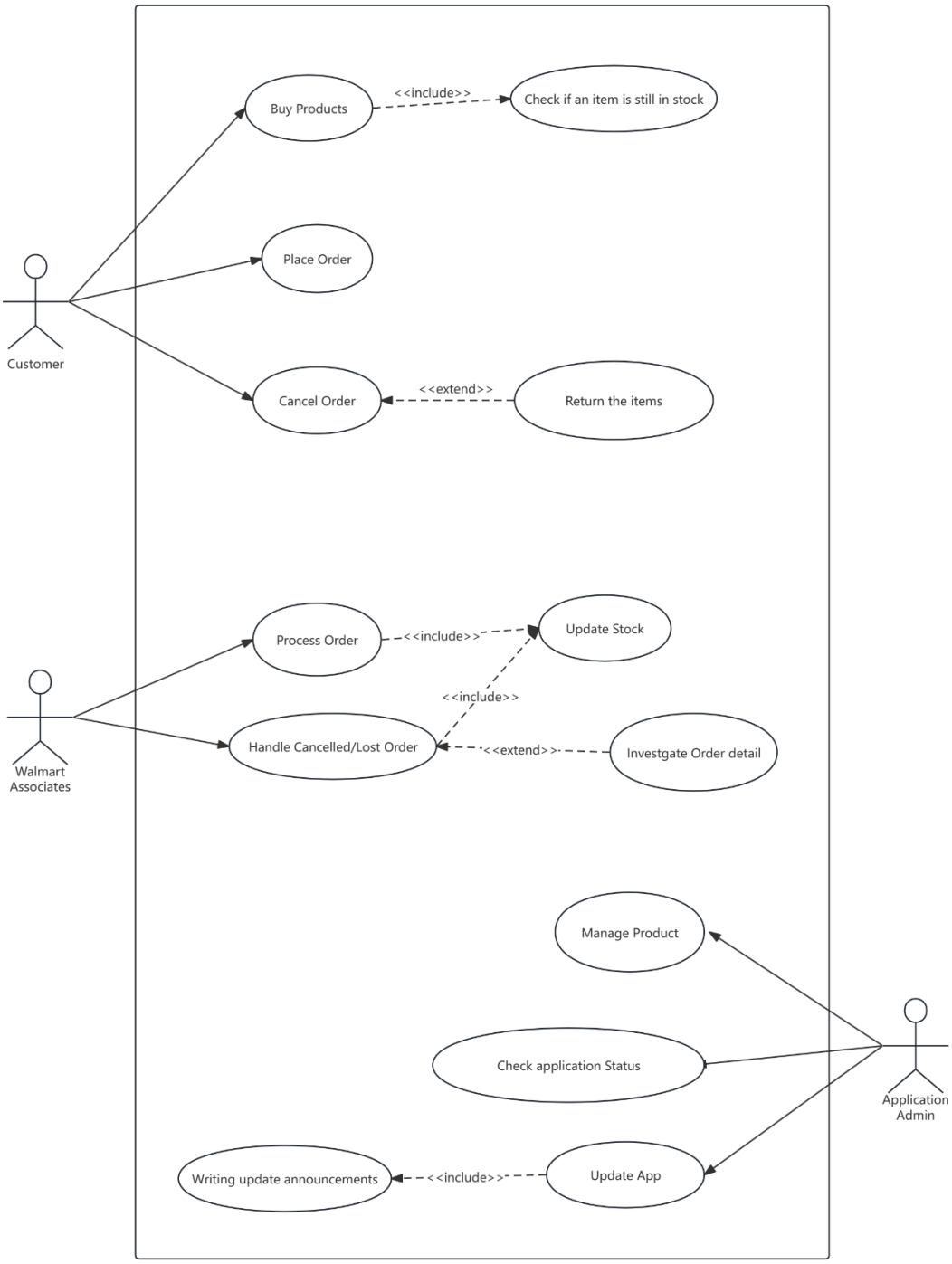
Q2. Consider the same Walmart application. Develop a **state diagram** representing the states of Walmart orders. **[30 points]**

The state machine diagram must specify all order states.



Q3. Develop a Use case Model for this application. Your actors for this model should include Customers, Walmart associates, and Application administrators. **[10 points]**

Your use case model should use **at least two include and two extend** relationships.



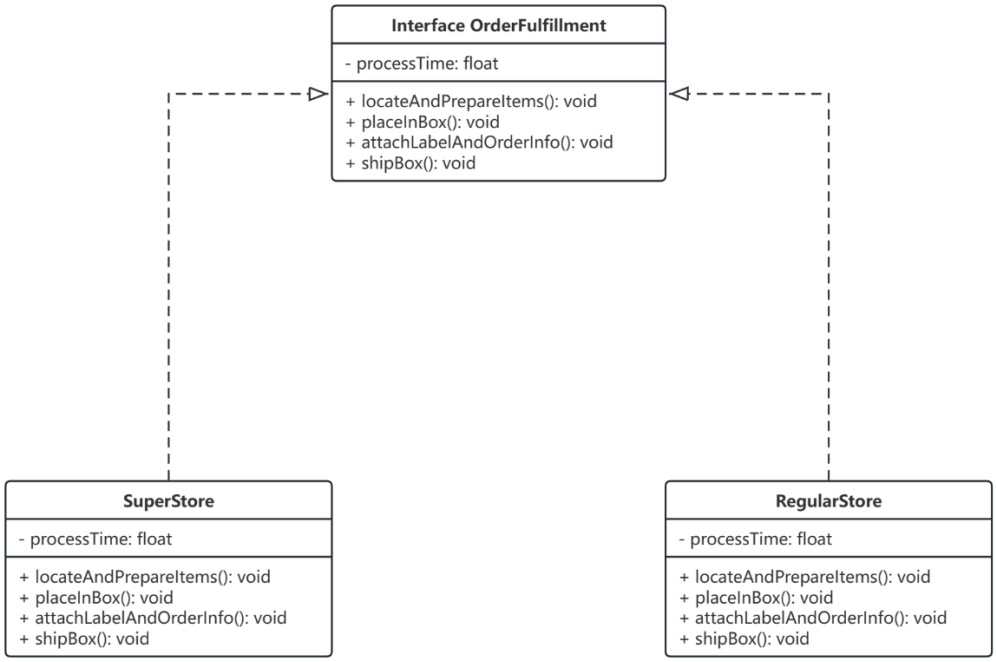
Q4. Choose one use case and develop a use case scenario, including all required elements. **[10 points]**

<table>
<tr><td colspan="2">Use Case：Process Order</td></tr>
<tr><td colspan="2">Description：Once an order is placed, Walmart associates prepare it for delivery or pick-up.
Actor：Walmart Associates
Precondition：Customer place order successfully.
Main success scenario:</td></tr>
<tr><td>User</td><td>System</td></tr>
<tr><td>2. View user order information.
3. Check if stock is still available and feedback order status.</td><td>1. Automatically assign orders to the nearest Walmart.
4. Update order status.</td></tr>
<tr><td colspan="2">Alternative: Automatically re-find another Walmart when an order is not processed after a certain amount of time or when an order is reported as unavailable.
Exceptional: There is a lack of stock or a need to transfer stock after a user has placed an order.</td></tr>
<tr><td colspan="2">Include Case Scenario:</td></tr>
<tr><td>User</td><td>System</td></tr>
<tr><td>Update stock information</td><td>Receive stock information and synchronize it to all Walmarts.</td></tr>
<tr><td colspan="2">Exceptional: More than one Walmart processing an item at the same time</td></tr>
</table>

Q5. Reconsider the same Walmart system description. From a Walmart perspective, each order requires certain activities to be fulfilled. Specifically, 1) items must be located and prepared, 2) items must be placed in an appropriate shipping box, 3) label and order information must be attached, 4) the box with the items must be shipped. Walmart superstores and regular stores can fulfill these orders, but superstores take a little more time to prepare the order.

Design an application for this aspect using Java interfaces.
Provide **1)** a UML diagram describing the design, and **2)** a Java code snippet showing how the design can be realized in code. **[20 points]**

1)



2)

OrederFulfillment.java

```java
public interface OrderFulfillment {
    float getProcessTime();

    void locateAndPrepareItems();

    void placeInBox();

    void attachLabelAndOrderInfo();

    void shipBox();
}
```

SuperStore.java

```java
// Implementation for SuperStores
public class SuperStore implements OrderFulfillment {

    private float processTime = 0.0f;
```

```
    @Override
    public float getProcessTime() {
        return processTime;
    }

    @Override
    public void locateAndPrepareItems() {
        // a little bit more time than RegularStore
        processTime += 2.5f;
    }

    @Override
    public void placeInBox() {
        processTime += 1.0f;
    }

    @Override
    public void attachLabelAndOrderInfo() {
        processTime += 0.5f;
    }

    @Override
    public void shipBox() {
        processTime += 1.0f;
    }
}
```

RegularStore.java

```
// Implementation for RegularStores
public class RegularStore implements OrderFulfillment {

    private float processTime = 0.0f;

    @Override
    public float getProcessTime() {
        return processTime;
    }

    @Override
    public void locateAndPrepareItems() {
        processTime += 2.0f;
    }

    @Override
    public void placeInBox() {
        processTime += 1.0f;
    }

    @Override
    public void attachLabelAndOrderInfo() {
        processTime += 0.5f;
    }

    @Override
    public void shipBox() {
```

```
            processTime += 1.0f;
        }
    }
```

<<End of Exam>>