CSE1322L Assignment 3 - Fall 2025

Introduction:

Everyday, the United States Postal Service (USPS) delivers tens of millions of mail pieces throughout its territory, with this number varying throughout the year (reaching nearly 30 million mail pieces daily during the peak of the holidays). Mail pieces can be almost anything, such as postcards, letters, flats (technical term for a large envelope), and parcels (technical term for a box).

Implicit in the process of delivering mail is also the process of accepting mail. The complete specifications of what can be mailed are quite complex, involving simple factors like requiring destination and return address, to the dimensions and weight of the envelope or packaging, to whether a specific cargo can be sent via air mail or only via land mail, or if the item can be mailed at all.

We shall simulate a system which takes in information about a mail piece and determines if it can be sent for delivery or not. Our simple system will handle 4 kinds of mail: letters, flats for carrying documents, regular boxes for carrying anything, and live boxes for carrying honeybees or chickens (see USPS Publication 526 Mailable Live Animals if you are curious about the real thing).

This assignment requires a UML diagram to be submitted along with your code. UML diagram entities must contain:

- **The name of the class**
- **All of the classes' fields and their access levels (types are optional)**
- **All of the classes' methods (their return types and parameter types are optional. Constructors are also optional)**
- **Arrows indicating the relationship between classes (i.e.: an arrow pointing from Class A to Class B means that Class B is the parent of Class A)**

Requirements

The features described below must be in your program.

- A total of eight classes: MailRoom, Mail, Envelope, Box, Letter, Flat, RegularBox, LiveBox
- Mail must have the following fields:
    - A string called "deliveryAddress"
    - A string called "returnAddress"
    - A double called "width"
    - A double called "length"
    - An integer called "id"
    - A static integer called "nextId", initialized at 0
- Mail must have the following constructors

    - **Mail()**: Initializes all strings to empty, all doubles to 0, assigns nextId to id, and increments nextId
    - **Mail(String, String, double, double)**: assigns the first 2 arguments to deliveryAddress and returnAddress and the next 2 arguments to width and length, respectively. It then assigns nextId to id, and increments nextId
- Mail must have getters for all fields except nextId
- Mail must override toString() to return a string in the following format (replace the curly braces and its contents with the information stored in the specified field):

```
{ID}
{DELIVERY ADDRESS}
{RETURN ADDRESS}
```

- Envelope must be a child of Mail
- Envelope must have field of type double called “thickness”
- Envelope must have the following constructors:
    - **Envelope()**: Calls its parent’s default constructor and initializes thickness to 0
    - **Envelope(String, String, double, double, double)**: Passes the first 4 arguments to its superclass overloaded constructor and uses the last argument to set thickness.
- Envelope must have a getter for thickness
- Envelope must override toString() to return a string in the following format:

```
{ID}
{DELIVERY ADDRESS}
{RETURN ADDRESS}
{WIDTH} x {LENGTH} x {THICKNESS}
```

- Letter is a child of Envelope.
    - Letter must have a string field called “letterBody”
- Letter must have the following constructors:
    - **Letter()**: Calls the parent default constructor and sets letterBody to empty.
    - **Letter(String, String, double, double, double, String)**: Passes the first 5 argument to its superclass overloaded constructor and uses the last argument to set letterBody.
- Letter must have a getter for letterBody
- Flat is a child of Envelope.
- Flat must have a string field called “contents”
- Flat must have the following constructors:
    - **Flat()**: Calls the default constructor of its superclass and sets contents to empty.
    - **Flat(String, String, double, double, double, String)**: Passes the first 5 arguments to its superclass constructor and then uses the last argument to set the contents field
- Flat must have a getter for contents.
- Box must be a child of Mail
- Box must have the following private fields:
    - A double called “height”

    - An int called “count”
- Box must have the following constructors:
    - **Box()**: Calls its superclass default constructor and then sets height and count to 0.
    - **Box(String, String, double, double, double, int)**: Passes the first 4 arguments to its superclass overloaded constructor, and then uses the last 2 arguments to set its own fields.
- Box must have getters for both of its fields.
- Box must override toString() to return a string in the following format:

```
{ID}
{DELIVERY ADDRESS}
{RETURN ADDRESS}
{WIDTH} x {LENGTH} x {HEIGHT}
```

- RegularBox is a child of Box
- RegularBox must have the following private fields:
    - A string called items
    - A double called weight
- RegularBox must have the following constructors:
    - **RegularBox()**: Calls its superclass default constructor and sets weight to 0 and items to empty
    - **RegularBox(String, String, double, double, double, int, String, double)**: Passes the first 6 arguments to its superclass overloaded constructor and then uses the last 2 arguments to set its own fields
- RegularBox must have a getters for both its fields
- RegularBox must override toString() to return a string in the following format:

```
{ID}
{DELIVERY ADDRESS}
{RETURN ADDRESS}
{WIDTH} x {LENGTH} x {HEIGHT}
Items: {ITEMS}
Count: {COUNT}
Weight: {WEIGHT}
```

- LiveBox is a child of Box
- LiveBox must have the following private fields:
    - A string called animal
    - An int called age
- LiveBox must have the following constructors:
    - **LiveBox()**: sets animal to empty and age to 0.
    - **LiveBox(String, String, double, double, double, int, String, int)**: passes the first 6 arguments to its superclass overloaded constructor and uses the last 2 arguments to set its own fields
- LiveBox must have a getters for both its fields
- LiveBox must override toString() to return a string in the following format:

```
{ID}
```

```
{DELIVERY ADDRESS}
{RETURN ADDRESS}
{WIDTH} x {LENGTH} x {HEIGHT}
Animal: {ANIMAL}
Count: {COUNT}
Age (Days): {AGE}
```

- MailRoom must have the following static methods:
    - **boolean handleLetter(Letter, ArrayList<Mail>)**: If the first argument is a valid Letter, appends said argument to the second argument and returns true; otherwise, returns false. A valid Letter has the following characteristics:
        - Has non-empty delivery and return addresses.
        - 5 in <= length <= 11.5 in
        - 3.5 in <= width <= 6.125 in
        - 0.007 <= thickness <= 0.25 in
        - Is rectangular (i.e.: the length is at least 1.5 inches longer than the width)
    - **boolean handleFlat(Flat, ArrayList<Mail>)**: If the first argument is a valid Flat, appends said argument to the second argument and returns true; otherwise, returns false. A valid Flat has the following characteristics:
        - Has non-empty delivery and return addresses.
        - 11.5 in <= length <= 15 in
        - 6.125 in <= width <= 12 in
        - 0.25 <= thickness <= 0.75 in
        - Can only have “DOCUMENTS” as its contents
    - **boolean handleRegularBox(RegularBox, ArrayList<Mail>)**: If the first argument is a valid RegularBox, appends said argument to the second argument and returns true; otherwise, returns false. A valid RegularBox has the following characteristics:
        - Has non-empty delivery and return addresses.
        - 6 in <= length <= 27 in
        - 0.25 in <= width <= 17 in
        - 3 <= height <= 17 in
        - 0 lbs <= weight <= 70 lbs
        - 0 <= count <= 50
    - **boolean handleLiveBox(LiveBox, ArrayList<Mail>)**: If the first argument is a valid LiveBox, appends said argument to the second argument and returns true; otherwise, returns false. A valid LiveBox has the following characteristics:
        - Has non-empty delivery and return addresses.
        - 6 in <= length <= 27 in
        - 0.25 in <= width <= 17 in
        - 3 <= height <= 17 in
        - Must be carrying either “HONEYBEES” or “CHICKEN”.
        - If carrying honeybees, must have no less than 0 and no more than 20 honeybees.
        - If carrying chicken, must have no less than 0 and no more than 10 chickens.

        - If carrying chicken, the age of the oldest chicken cannot be greater than 1 day or negative.
    - **main()**: the main method
- MailRoom's main method must do the following:
    - Create an ArrayList of Mail called "deliver"
    - Have a menu in a loop which prompts the user for what they wish to do
    - If the user tries to mail something, the menu must prompt them for the appropriate information to create an object of that mail type. It must then call the appropriate static method and pass to it both the newly created mail object and the arraylist, and then inform the user if their mail object is valid or not.
        - For example, if the user wishes to send a letter, prompt them for the delivery address, the return address, the width, the length, the thickness, and the contents of the letter. Then, create a Letter object with the information above and pass it to handleLetter() along with the arraylist.
    - The menu must have an option to deliver the mail, whereupon it prints all of the objects in the arraylist and then empties the arraylist.
    - The menu must have an option to end the program.

Deliverables

- MailRoom.java (driver)
- Mail.java
- Envelope.java
- Box.java
- Letter.java
- Flat.java
- RegularBox.java
- LiveBox.java
- UML.pdf

Considerations

- This assignment may seem intimidating, but that's just because of the number of things you have to do; the assignment itself isn't very hard, so don't be discouraged.
- Remember that you will get partial credit for partial work. Try to deliver as much of the assignment as you can.
- You may add any helper methods you believe are necessary, but you will not get points for them.
- Despite what the deliverables above say, you can submit all of your classes in a single file.
- Remember to make full use of inheritance to avoid writing the same lines of code multiple times.
- While you may use any software to generate your UML diagram (or even submit a hand drawn version), it is best to use the recommended plugin in order to avoid missing any information that the diagram requires. Incorrect diagrams will lose you points and illegible ones will not be graded.

Sample Output (user input in red)

```
Welcome to KSUPS

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 1

Enter the delivery address: 1000 Chastain Road, Kennesaw, GA, 30144
Enter the return address: 1100 South Marietta Pkwy, Marietta, GA, 30060
Enter the width of your mail: 5
Enter the length of your mail: 7
Enter the thickness of your mail: 0.2
Enter the contents of your letter: Game Conference Invitation

Letter accepted for delivery.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 2

Enter the delivery address: 1000 Chastain Road, Kennesaw, GA, 30144
Enter the return address: 1100 South Marietta Pkwy, Marietta, GA, 30060
Enter the width of your mail: 10
Enter the length of your mail: 14
Enter the thickness of your mail: 0.5
Enter the contents of your Flat: DOCUMENTS

Flat accepted for delivery.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
```

```
Enter option: 3

Enter the delivery address: 1000 Chastain Road, Kennesaw, GA, 30144
Enter the return address: 1100 South Marietta Pkwy, Marietta, GA, 30060
Enter the width of your mail: 10
Enter the length of your mail: 10
Enter the height of your mail: 10
Enter how many items your box has: 2
Enter the weight of your box: 50
Enter the contents of your box: Statues

Regular box accepted for delivery.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 4

Enter the delivery address: 1000 Chastain Road, Kennesaw, GA, 30144
Enter the return address: 1100 South Marietta Pkwy, Marietta, GA, 30060
Enter the width of your mail: 7
Enter the length of your mail: 7
Enter the height of your mail: 7
Enter how many items your box has: 5
Enter what animal is being shipped: chicken
Enter age of the oldest animal, in days: 1

Live box accepted for delivery.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 3

Enter the delivery address: Local Orphanage
Enter the return address:
Enter the width of your mail: 15
```

```
Enter the length of your mail: 15
Enter the height of your mail: 15
Enter how many items your box has: 1
Enter the weight of your box: 60
Enter the contents of your box: Big Bomb

Regular box cannot be mailed.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 4

Enter the delivery address: 1000 Chastain Road, Kennesaw, GA, 30144
Enter the return address: 1100 South Marietta Pkwy, Marietta, GA, 30060
Enter the width of your mail: 12
Enter the length of your mail: 12
Enter the height of your mail: 12
Enter how many items your box has: 20
Enter what animal is being shipped: honeybees
Enter age of the oldest animal, in days: 200

Live box accepted for delivery.

1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 5

DISPATCHING THE FOLLOWING ITEMS FOR DELIVERY:
==========
0
1000 Chastain Road, Kennesaw, GA, 30144
1100 South Marietta Pkwy, Marietta, GA, 30060
5.0 x 7.0 x 0.2
==========
1
```

```
1000 Chastain Road, Kennesaw, GA, 30144
1100 South Marietta Pkwy, Marietta, GA, 30060
10.0 x 14.0 x 0.5
==========
2
1000 Chastain Road, Kennesaw, GA, 30144
1100 South Marietta Pkwy, Marietta, GA, 30060
10.0 x 10.0 x 10.0
Items: Statues
Count: 2
Weight: 50.0
==========
3
1000 Chastain Road, Kennesaw, GA, 30144
1100 South Marietta Pkwy, Marietta, GA, 30060
7.0 x 7.0 x 7.0
Animal: CHICKEN
Count: 5
Age (Days): 1
==========
5
1000 Chastain Road, Kennesaw, GA, 30144
1100 South Marietta Pkwy, Marietta, GA, 30060
12.0 x 12.0 x 12.0
Animal: HONEYBEES
Count: 20
Age (Days): 200
==========


1- Send letter
2- Send flat
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 5

Delivery queue is empty.


1- Send letter
2- Send flat
```

```
3- Send regular box
4- Send live box
5- Dispatch items
6- Quit
Enter option: 6

Quitting...
```