

Module 8 - Inheritance and Polymorphism

CMPT220L

Due on Nov 6, 2020 by 11:59 PM

Points: 100

Problems

1. (*Split a string*) Write the following method that splits a string into substrings using delimiter characters.

```
public static ArrayList<String> split(String s, String delimiters);
```

For example, `split("AB#C D?EF#45", "# ?")` returns a vector containing AB, C, D, EF, and 45. Write a test program that prompts the user to enter a string and delimiters and displays the substrings separated by a space. (**You are not allowed to use the regex for splitting a string in this exercise.**)

2. (*Bin packing with largest object first*) The bin packing problem is to pack the objects of various weights into containers. Assume that each container can hold a maximum of 10 pounds. The program uses an algorithm that places an object with the largest weight into the first bin in which it would fit. Your program should prompt the user to enter the total number of objects and the weight of each object. The program displays the total number of containers needed to pack the objects and the contents of each container. Here is a sample run of the program:

```
Enter the number of objects: 6
Enter the weights of the objects: 7 5 2 3 5 8
Container 1 contains objects with weight 8 2
Container 2 contains objects with weight 7 3
Container 3 contains objects with weight 5 5
```

3. (*Flight time*) Design two classes: `Flight` and `Itinerary`. The `Flight` class stores the information about a flight with the following members:
 - (a) A data field named `flightNo` of the `String` type with getter method.
 - (b) A data field named `departureTime` of the `GregorianCalendar` type with getter and setter methods.
 - (c) A data field named `arrivalTime` of the `GregorianCalendar` type with getter and setter methods.
 - (d) A constructor that creates a `Flight` with the specified `number`, `departureTime`, and `arrivalTime`.
 - (e) A method named `getFlightTime()` that returns the flight time in minutes.

The `Itinerary` class stores the information about itinerary with the following members:

- (a) A data field named `flights` of the `List<Flight>` type. The list contains the flights for the itinerary in increasing order of `departureTime`.
- (b) A constructor that creates an `Itinerary` with the specified flights.
- (c) A method named `getTotalTime()` that returns the total travel time in minutes from the departure time and the first flight to the arrival time of the last flight in the itinerary.

Implement these two classes and use the following program to test these classes.

```

public static void main(String[] args) {
    List<Flight> flights = new ArrayList<>();
    flights.add(new Flight("US230",
        new GregorianCalendar(2014, 4, 5, 5, 5, 0),
        new GregorianCalendar(2014, 4, 5, 6, 15, 0)));
    flights.add(new Flight("US235",
        new GregorianCalendar(2014, 4, 5, 6, 55, 0),
        new GregorianCalendar(2014, 4, 5, 7, 45, 0)));
    flights.add(new Flight("US237",
        new GregorianCalendar(2014, 4, 5, 9, 35, 0),
        new GregorianCalendar(2014, 4, 5, 12, 55, 0)));

    Itinerary itinerary = new Itinerary(flights);
    System.out.println(itinerary.getTotalTravelTime())System.out.println(itinerary.getTotalFlightTime());
}

```

Submission

Make sure you create one Java file per project. Place your .java files under the corresponding folder in your local copy of the GitHub repository, commit and push it to the remote repository. Make sure that the professor has access to the repository (jfac65-marist).

```

cmpt220lastname\
  hw\
    8\
      Problem1.java
      Problem2.java
      Problem3.java

```