

3. Based on the following UML class diagram in Figure 8.1, write a complete C++ program that implements the relationships among classes. The explanation of each method is as in Table 8.2.

Table 8.2: Explanation of each method in each class

Methods	Explanation
<b>Flight class</b>	
<b>addPassenger (Passenger)</b>	This method will add <b>Passenger</b> objects to the <b>passengerList</b> vector.
<b>printInfo ()</b>	This method will display all flight information as an example below:  Flight No: MH3120 Destination: Johor Bahru Departure: 8:10 Arrival: 9:00 Number of Passengers: 10

<b>Passenger</b> <b>printDetails()</b>	This method will display all passenger information, i.e. <b>name</b> and <b>age</b> .
<b>Kids class</b> <b>printDetails()</b>	<ul style="list-style-type: none"> <li>This method will display all kids passenger information.</li> <li>This method overrides its superclass <b>printDetails</b> method. The values of its inherited attributes will be displayed by calling its inherited method.</li> </ul>
<b>Adult class</b> <b>printDetails()</b>	<ul style="list-style-type: none"> <li>This method will display all adults passenger information.</li> <li>This method overrides its superclass <b>printDetails</b> method. The values of its inherited attributes will be displayed by calling its inherited method.</li> </ul>
<b>Time class</b> <b>getHour()</b>	This method will return the attribute value of <b>hour</b> .
<b>getMinute()</b>	This method will return the attribute value of <b>minute</b> .

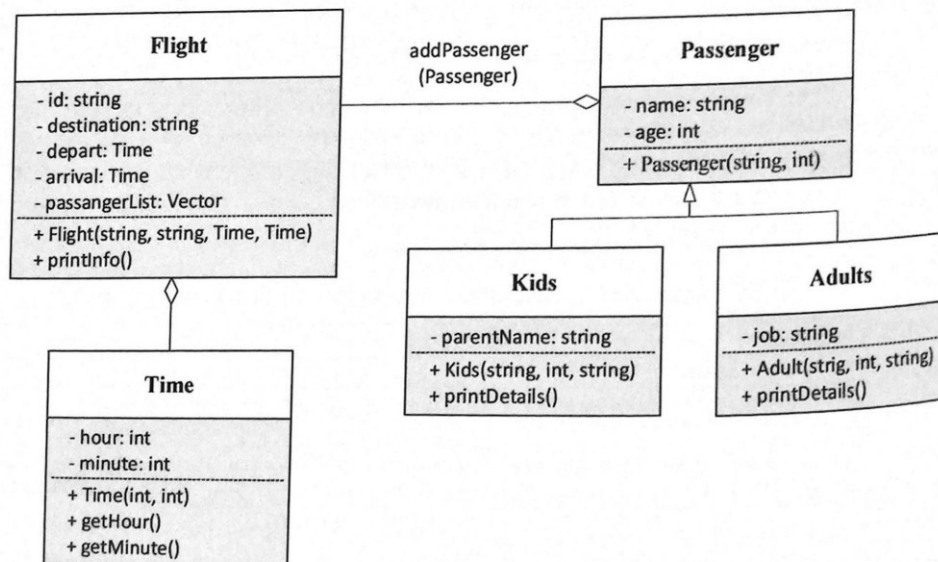


Figure 8.1: Class Diagram of **Flight**, **Passenger**, **Kids**, **Adult** and **Time** classes

Your program must be able to produce the following output as illustrated in Figure 8.2.

```
Flight Number: NAS7921
Destination:   Johore Bahru
Departure:    8:10
Arrival:      9:00

Number of Passengers: 3
Number of Adults:    2
Number of Kids:      1

Passengers Details:
Name:  Ali    Age:  34    Job Title    Lecturer
Name:  Goh    Age:  25    Job Title    Student
Name:  Azmi   Age:   3    Parents Name: Ali
```

**Figure 8.2:** Example of expected output