CENG 241
OBJECT ORIENTED PROGRAMMING

Flight Reservation System

Group No: 2

202011405 Pelin Su GÖK
201911009 Ahmet Berkay ASLAN

TABLE OF CONTENTS

1. INTRODUCTION

Program Language: C++

Main Programs Name: Flight Reservation System

Compiler: CLion and DevC++

A flight reservation system is a software used for managing transportation for flight companies.

1.1 Purpose

The purpose of our program is to provide the customers easy and understandable system for their flight reservation.

We aim to create a simple reservation system for customers easy use. The program provides its customers to choose their flights, flight packages, and their seats. In the program there are 3 different types of flight packages, each package has its price.

The program also provides a discount for children.

1.2 Scope

We created a system plan by analyzing some other flight company's websites, to figure out what data the system needs. Then we created a class diagram to figure out how the system is going to work. Then we transfer our algorithm and class diagram into code by using C++. Last, of all, we tested our program many times for testing the accuracy of our program.

2. DESIGN

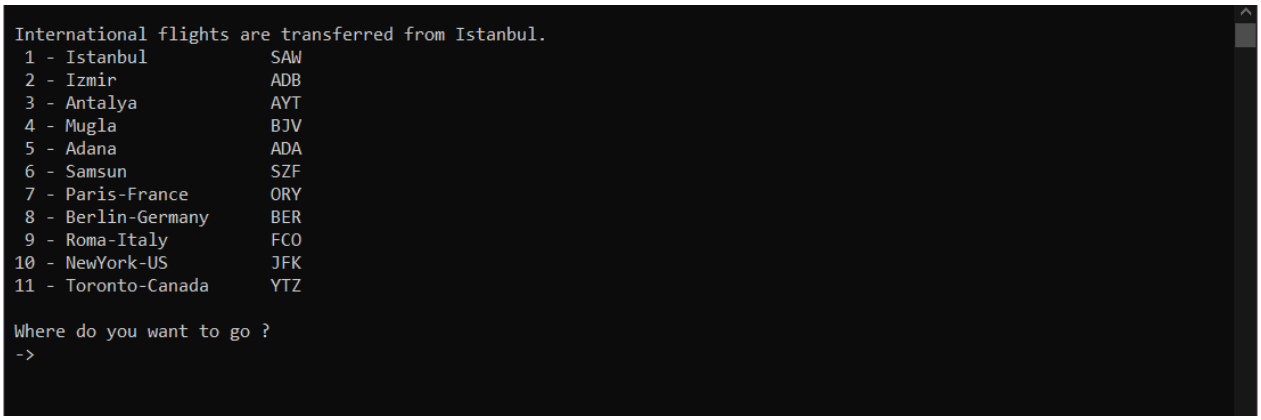
2.1 Approach

Our program is based on a simple and easily understandable menu. Information that is needed for the user to buy a ticket is very clear in the program.

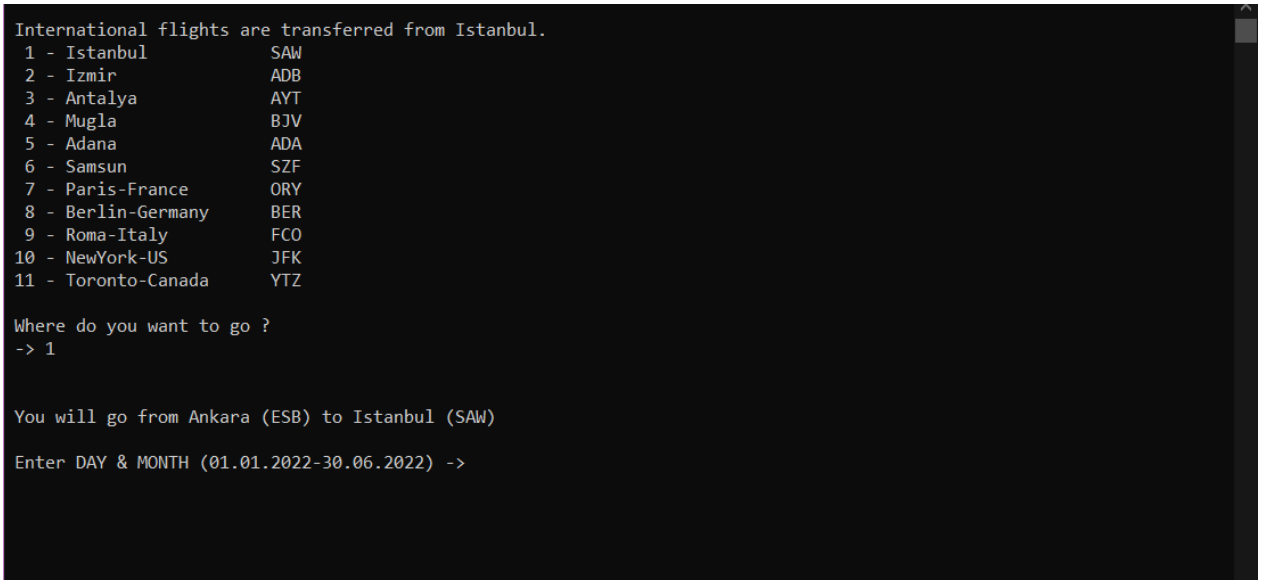
2.2 System

With this program, users can buy a ticket, check their ticket, choose where they want to go, choose when they want to go. The menu system works by the user to input their choice, generally by numbers.

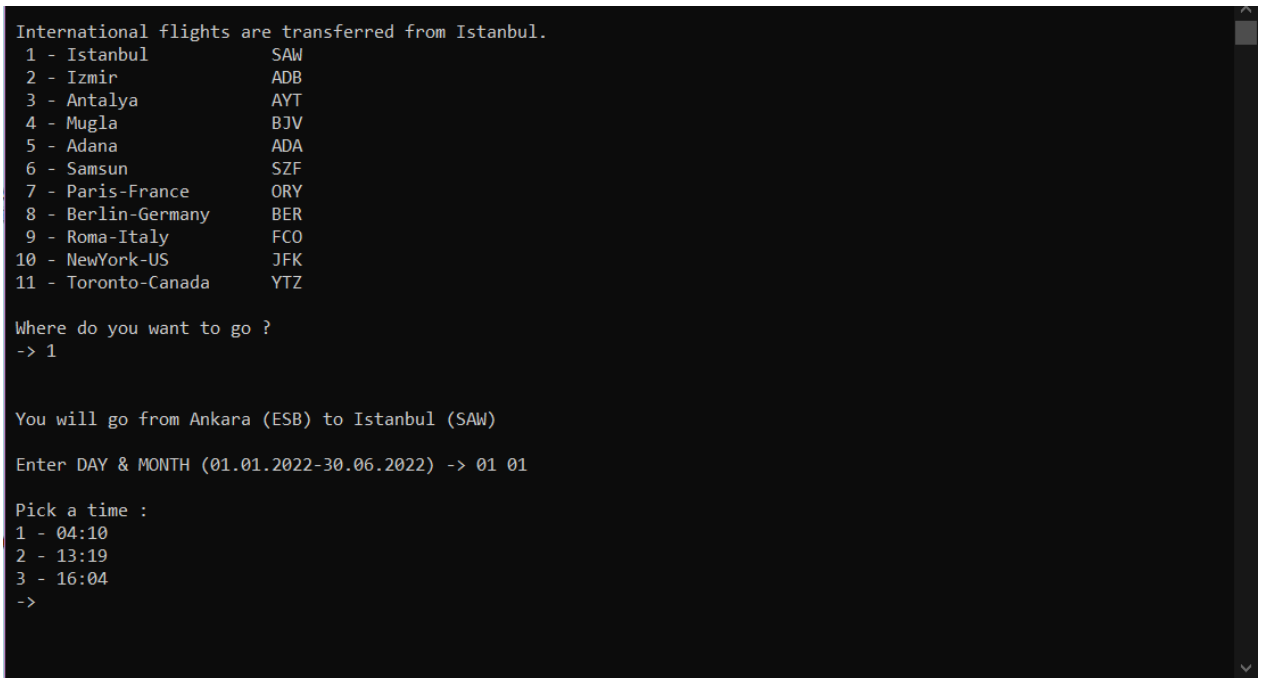
When the user wants to buy the ticket the program will print the flights on its system. And ask the user where they want to go:



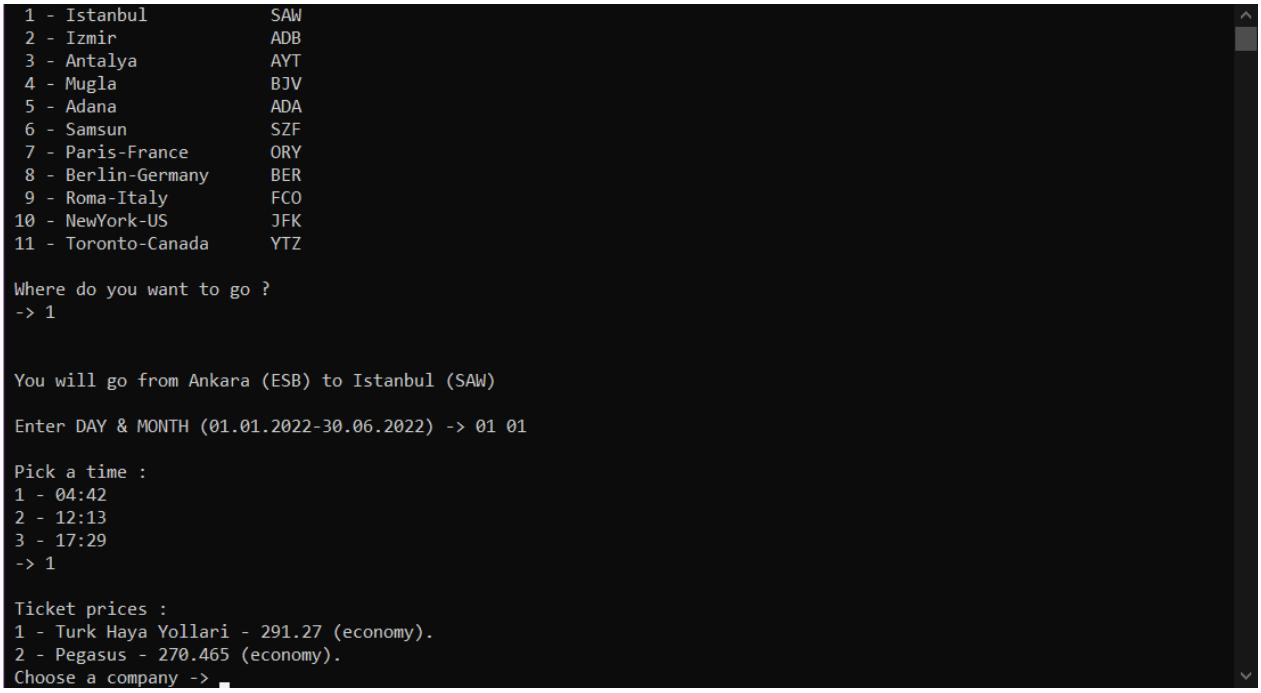
When the user chooses their destination, the program will ask the user when they want to travel:



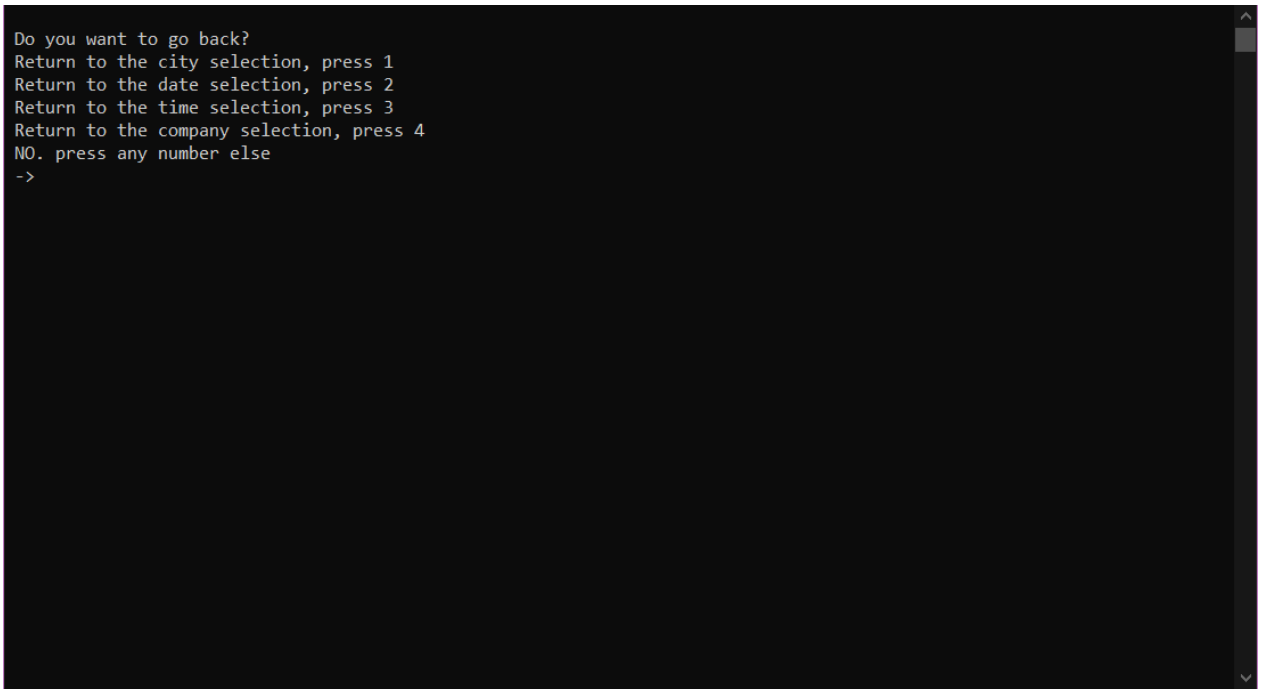
Then the program asks the user what time do they want to travel:



Then the program asks the user which company do they prefer:



After choosing the company program offers an exit, in case of any wrong input by the user:



If the user wants to continue with purchasing, seats will be printed and occupied seats fill shown with an asterisk next to the seat number, then ask the user how many seats do they want to buy:

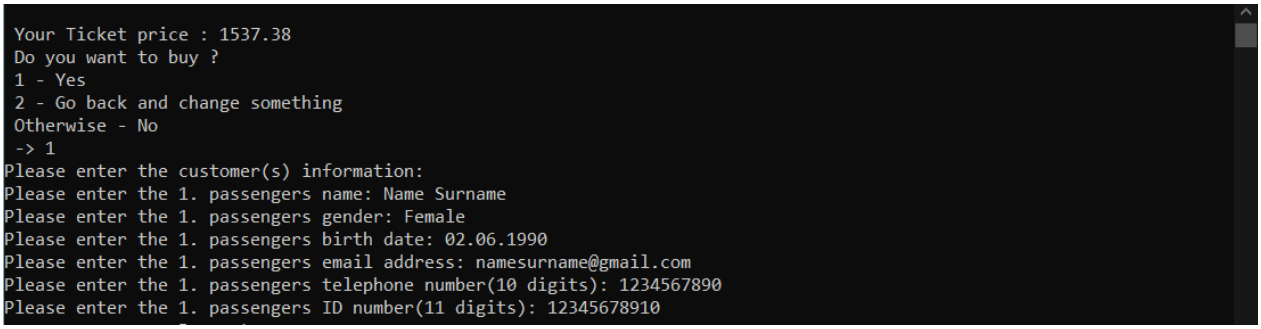
Then the program asks the user which seat do they want to buy, if user chooses occupied seats, program gives an error and ask user to choose a seat again:

Based on how many seats does the user wants to buy, the program asks how many children are going to be passengers in economy class:

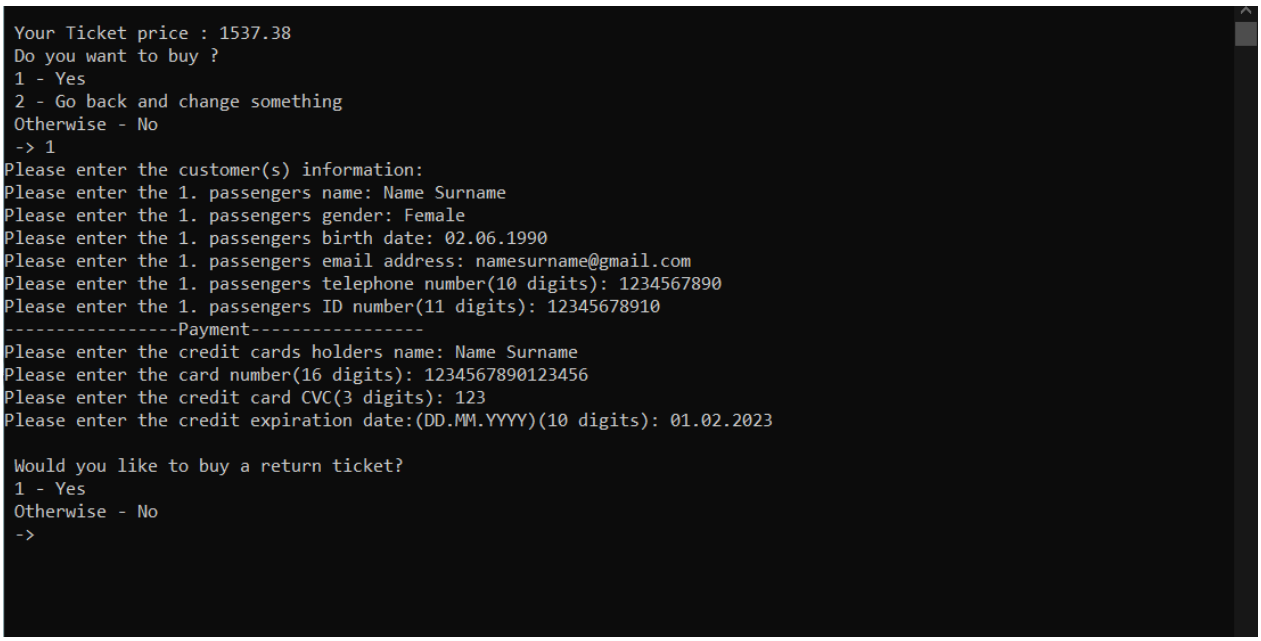
```
****** VIP Seats ******
1*   2*       3    4
5    6*       7    8*
9*   10*      11*  12

*** Business Seats ****
13   14*      15   16
17*  18*      19*  20
21*  22       23*  24*
25*  26*      27*  28*
29*  30       31*  32*

*** Economy Seats *****
33*  34*      35*  36
37*  38       39*  40
41*  42*      43*  44
45*  46*      47*  48*
49*  50*      51*  52
53*  54*      55*  56*
57*  58       59*  60*
61   62*      63*  64
65*  66*      67*  68*
69*  70*      71*  72*
73*  74       75   76
77   78*      79*  80*
81*  82       83*  84
85*  86*      87*  88
89*  90       91   92

How many seats do you want to buy -> 4


Choose seats -> 3 13 75 76

How many children? (children must be accompanied by at least one adult):
There are not enough seats for a child in vip seats.

There are not enough seats for a child in bussines seats.

How many children in economy -> 1
```

Then based on their seat number(which is connected to the flight package), the number of children, and the company of choice ticket price will calculate, and ask the user if they want to purchase the ticket or not:

If the user chooses to buy the ticket, program asks the user their personal information:



After the user inputs their personal information, the program asks for their payment information, and whether or not do the user want to purchase a return ticket or not:



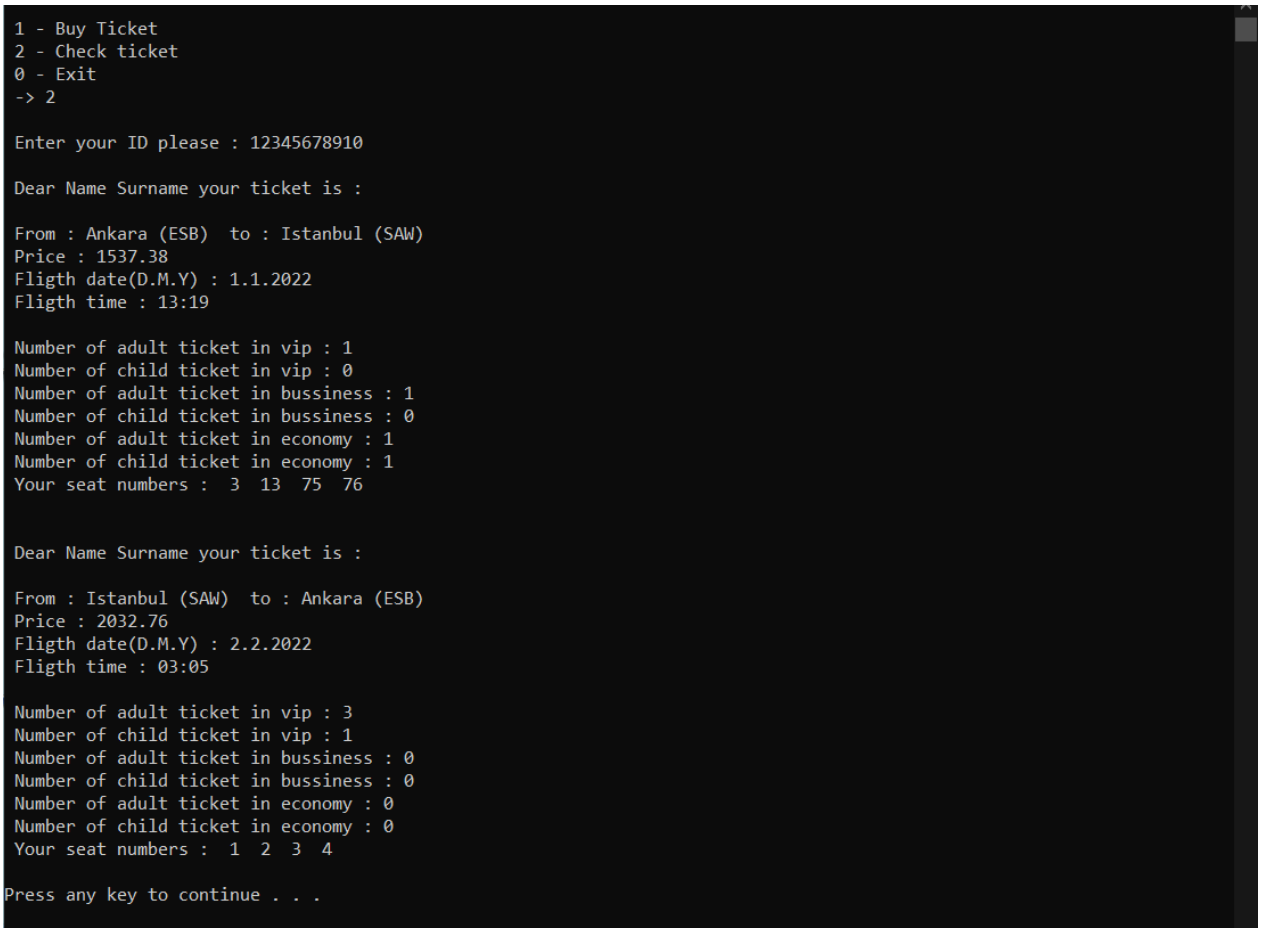
If the user chooses to buy a return ticket, the program prints a similar menu to before:

```
Enter DAY & MONTH (2.1.2022-30.06.2022) ->02 02

Pick a time :
1 - 03:05
2 - 12:40
3 - 16:35
-> 1

Ticket prices :
1 - Turk Haya Yollari - 260.61 (economy).
2 - Pegasus - 241.995 (economy).
Choose a company -> 1
```

```
****** VIP Seats ******
1    2        3    4
5    6        7    8

*** Business Seats ****
9    10       11   12
13*  14       15   16*
17   18       19   20

*** Economy Seats *****
21   22       23*  24
25*  26       27   28
29   30       31   32
33   34       35   36
37   38       39*  40
41   42       43   44
45   46       47   48
49   50*      51*  52
53   54       55   56

How many seats do you want to buy -> 4


Choose seats -> 1 2 3 4

How many children? (children must be accompanied by at least one adult):
How many children in vip -> 1
```

If the user wants to check their ticket, the program asks for their ID number:

When the user enters their ID number, their flight information is printed:

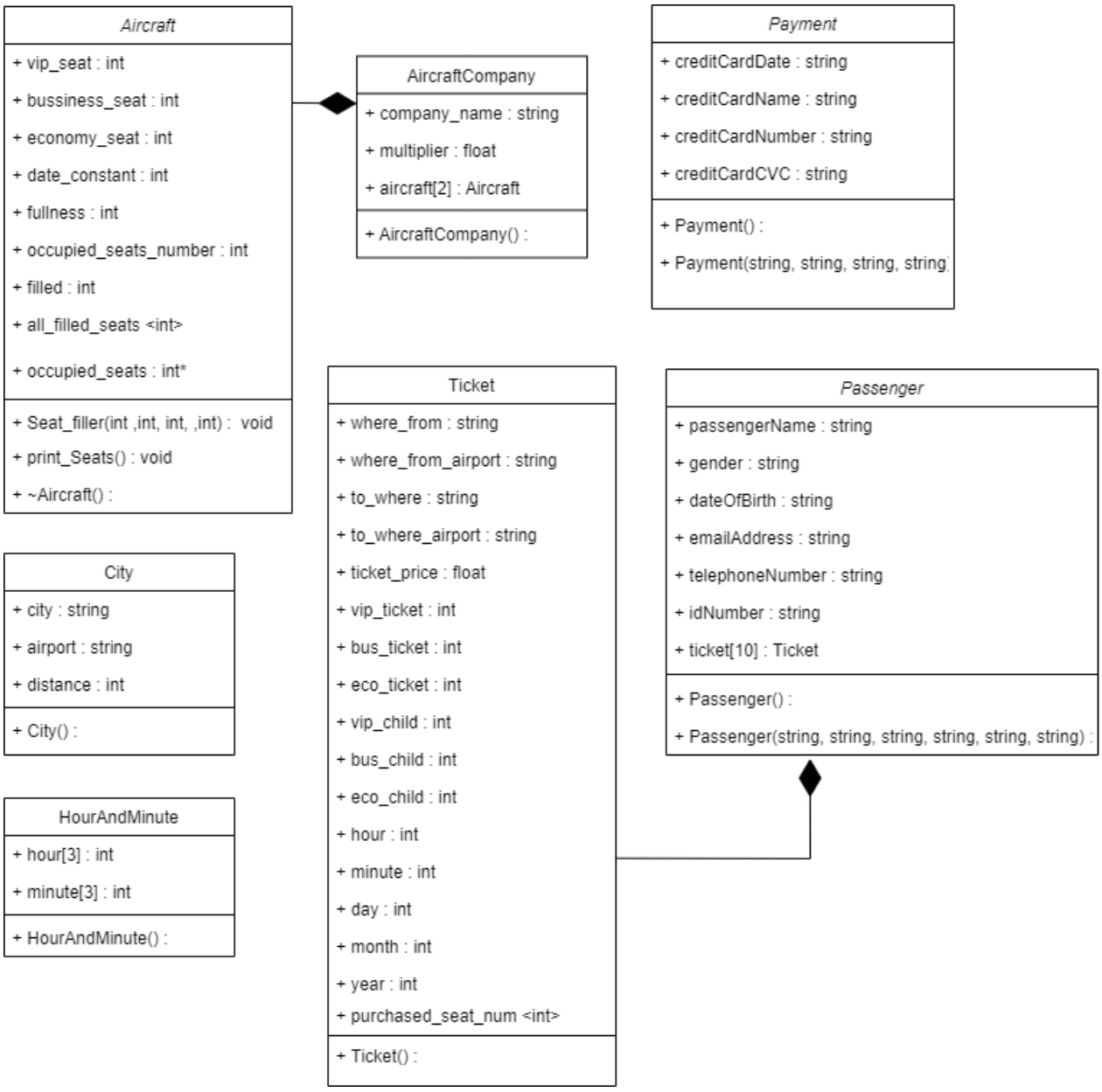


If the user wants to exit the program:

```
1 - Buy Ticket
2 - Check ticket
0 - Exit
-> 0


BYE !
```

2.3 System Design

Class Diagram:



3. CONCLUSION

This flight reservation system, was written with the needs of the users in mind. In this program, we have developed it by taking the advanced systems used today as an example. It was prepared according to the topics covered in the Ceng 241 course (functions, classes, vectors, dynamic memory, etc.) and it has been developed by using C++ (object-oriented) language. We believe that, the errors and warnings while writing this program that we encountered, will make us a better program developer in business life.