CENG 241

OBJECT ORIENTED PROGRAMMING

Vacation Reservation System

Team Members: Ahmet Utku Gökmen - 201911402

Doğa Melis Erke - 201911026

Özgün Doğan - 201911024

TABLE OF CONTENTS

Introduction of Program

Program Language: C++

Main Programs Name: Vacation Reservation System

Compiler: Visual Studio

Purpose

Most of the people want to make decisions quickly and comfortably. Therefore, our aim is to reduce their options and make their vacation reservations easier. Thus, people can make their decisions more easily thanks to algorithm that we created.

Scope

Within the scope of the project, our users have options and these options go from general to private. At first, they choose the vacation type, whether it is a camp, hotel or daily, and then user choose the season of vacation. For example, if user choose a hotel, s/he can select the name of the hotel user want to go to or if camp is chosen, s/he can select the name of the camp area. Similarly, user want to make a daily vacation. Afterwards, we get the number of contacts and get personal information from the user as many as the number of people. At the end, we are simulating the Payment. All this information is kept in the Reservation Class so that we can easily access all choices that user made.

System Design

- RESERVATION

In the reservation system, we have a “Reservation” class which includes all classes that are related with reservation. Those are vacation types (Hotels, Camps, Daily Trips) and customer types (Adult, Elder, Baby, Student) as arrays. Thanks to this class we keep all the choices user made in that class and we can access them. Moreover, this class has an operation overload that is output function “<<”. Thanks to operator overload we are able to print all the information user made. Reservation class includes “=” operator also and it provide us copy all choices that are made by user to suitable variables in that class.

- OBJECTS

- Vacation Types

We create objects from text files. We have 3 different text files for each vacation type classes (hotel, camp and daily trip) that are inherited from one parent class named as “VacationTypes”. After program starts, compiler creates all objects with functions that exist in main file thanks to “=” operator overload and global vectors we can manipulate property of hotels or camps or daily trip and put them in a suitable global vector. Therefore, according to preferences of user our algorithm eliminates some objects if choice does not match with property of objects. As an example, if user wants to choose a hotel that has a season type as winter, we show him/her only name of winter hotels with type, price and place.

- Customer Types

We have 4 customer type class “Adult”, “Elder”, “Student”, “Baby”. Each class are inherited from one abstracted parent class named as “Customer”. Discount function in parent class is pure virtual. Usage of polymorphism provides us to apply discount to a total price, that user must pay, with different ratios.

- Main Function

In main user is met with decision tree (can be seen in figure “Decision Tree”) that consist of switch case method. Right after the selections reservation information is created (Starting Date, Vacation type, e.g.). We control most of the step of our user with do while loop or try catch methods and error massage is printed in them. Not only we are printing an error massage but also, we force user input a valid value. Furthermore, while subclasses of “Customer” class is creating, we check validity of user inputs and applying same procedure as well. All choices are saving in “Reservation” class. Finally, payment simulation occurs and program ends.

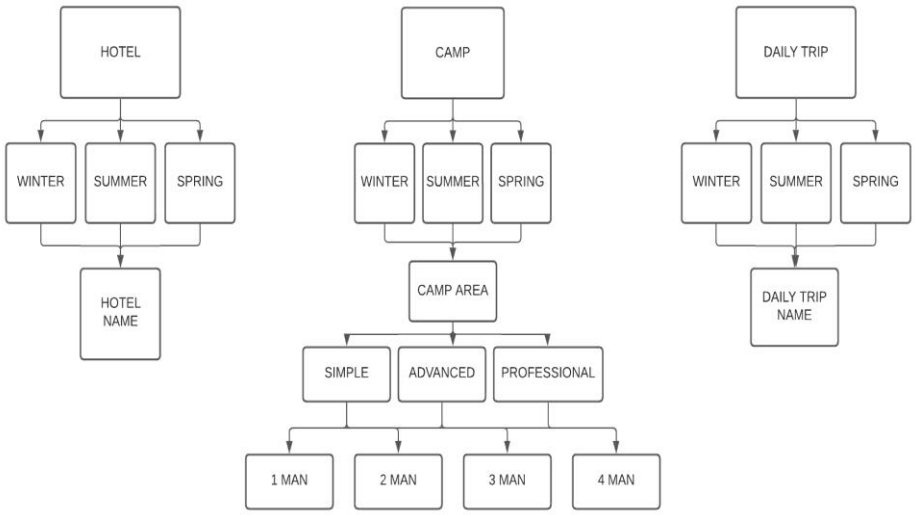
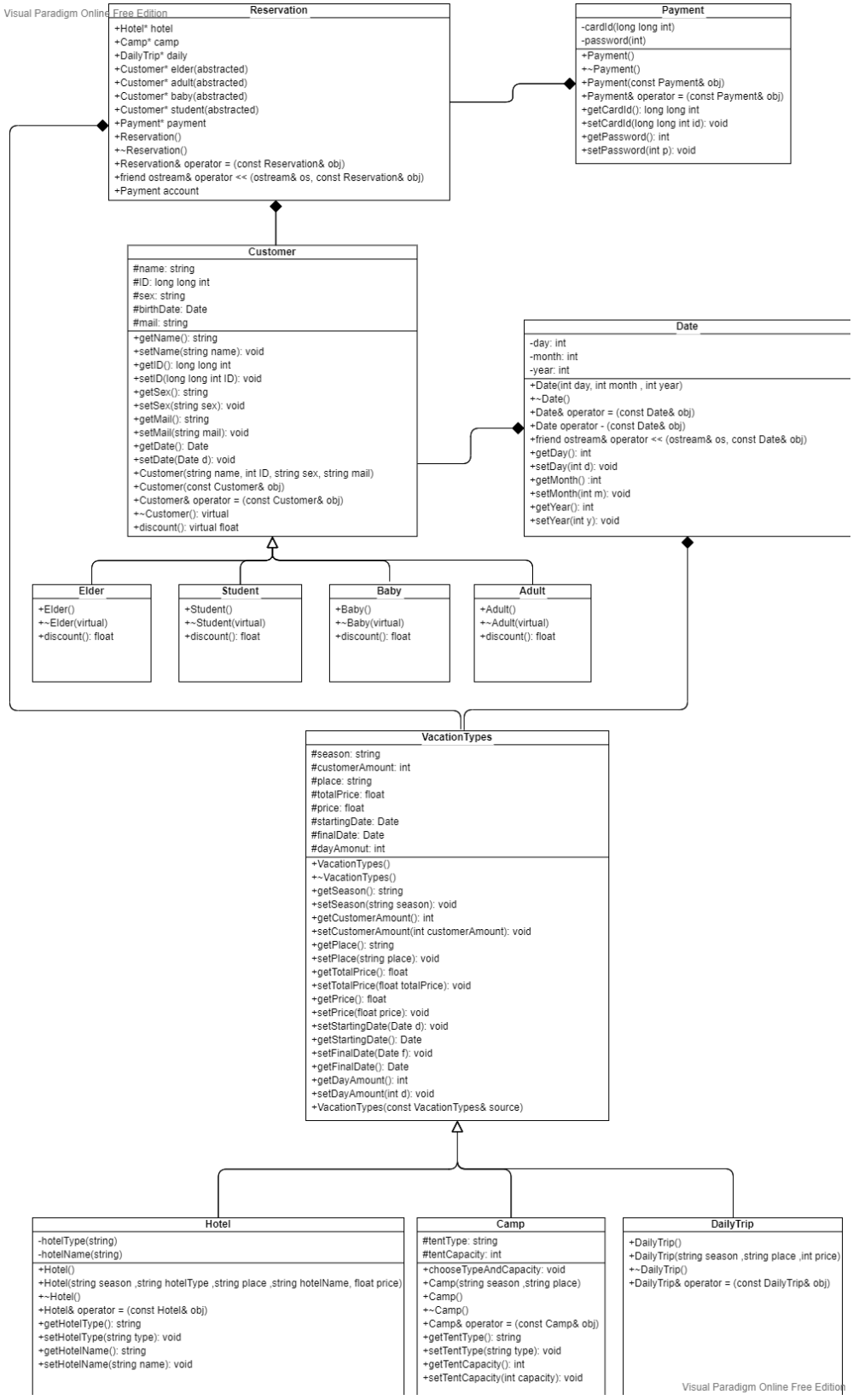
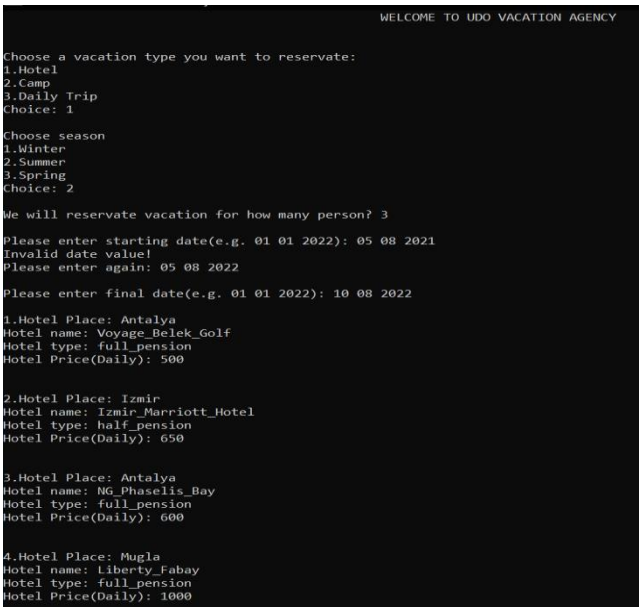


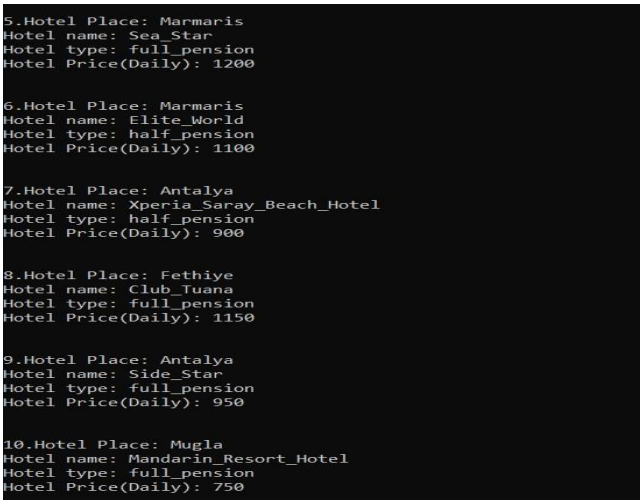
Figure Decision Tree

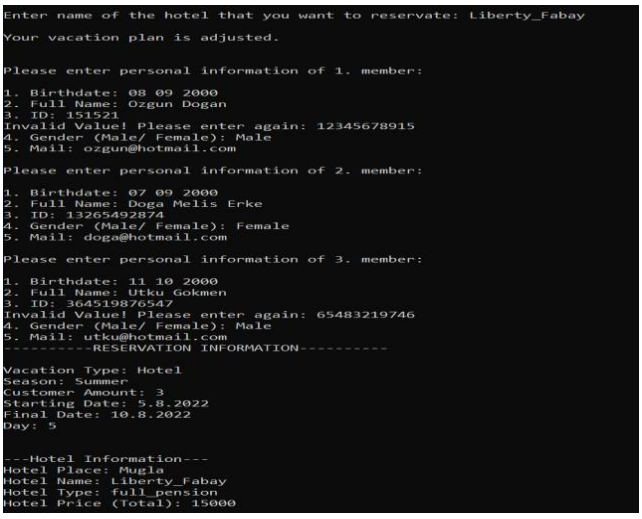
Class Diagram

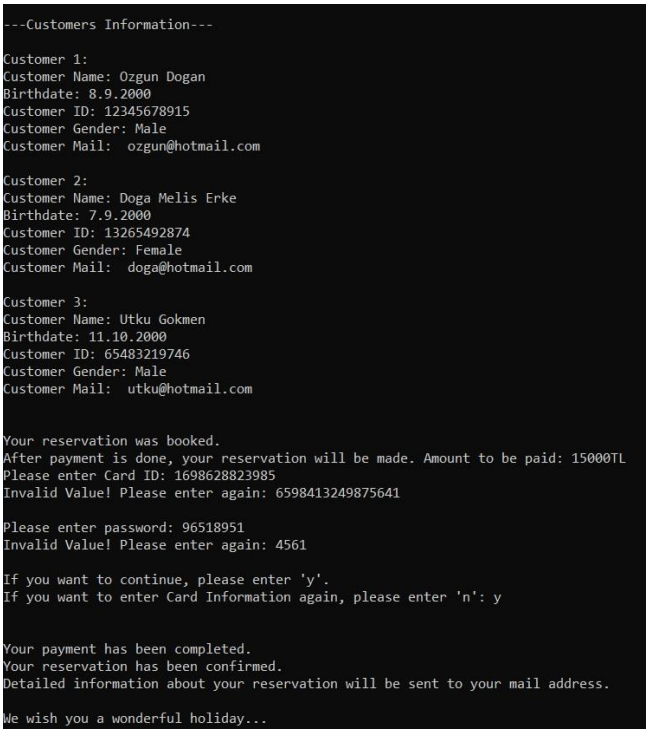


Sample Run


WELCOME TO UDO VACATION AGENCY
Choose a vacation type you want to reservate:
1.Hotel
2.Camp
3.Daily Trip
Choice: 1
Choose season
1.Winter
2.Summer
3.Spring
Choice: 2
We will reservate vacation for how many person? 3
Please enter starting date(e.g. 01 01 2022): 05 08 2021
Invalid date value!
Please enter again: 05 08 2022
Please enter final date(e.g. 01 01 2022): 10 08 2022
1.Hotel Place: Antalya
Hotel name: Voyage_Belek_Golf
Hotel type: full_pension
Hotel Price(Daily): 500
2.Hotel Place: Izmir
Hotel name: Izmir_Marriott_Hotel
Hotel type: half_pension
Hotel Price(Daily): 650
3.Hotel Place: Antalya
Hotel name: NG_Phaselis_Bay
Hotel type: full_pension
Hotel Price(Daily): 600
4.Hotel Place: Mugla
Hotel name: Liberty_Fabay
Hotel type: full_pension
Hotel Price(Daily): 1000


5.Hotel Place: Marmaris
Hotel name: Sea_Star
Hotel type: full_pension
Hotel Price(Daily): 1200
6.Hotel Place: Marmaris
Hotel name: Elite_World
Hotel type: half_pension
Hotel Price(Daily): 1100
7.Hotel Place: Antalya
Hotel name: Xperia_Saray_Beach_Hotel
Hotel type: half_pension
Hotel Price(Daily): 900
8.Hotel Place: Fethiye
Hotel name: Club_Tuana
Hotel type: full_pension
Hotel Price(Daily): 1150
9.Hotel Place: Antalya
Hotel name: Side_Star
Hotel type: full_pension
Hotel Price(Daily): 950
10.Hotel Place: Mugla
Hotel name: Mandarin_Resort_Hotel
Hotel type: full_pension
Hotel Price(Daily): 750


Enter name of the hotel that you want to reservate: Liberty_Fabay
Your vacation plan is adjusted.
Please enter personal information of 1. member:
1. Birthdate: 08 09 2000
2. Full Name: Ozgun Dogan
3. ID: 151521
Invalid Value! Please enter again: 12345678915
4. Gender (Male/ Female): Male
5. Mail: ozgun@hotmail.com
Please enter personal information of 2. member:
1. Birthdate: 07 09 2000
2. Full Name: Doga Melis Erke
3. ID: 13265492874
4. Gender (Male/ Female): Female
5. Mail: doga@hotmail.com
Please enter personal information of 3. member:
1. Birthdate: 11 10 2000
2. Full Name: Utku Gokmen
3. ID: 36451987654
Invalid Value! Please enter again: 65483219746
4. Gender (Male/ Female): Male
5. Mail: utku@hotmail.com
----------RESERVATION INFORMATION----------
Vacation Type: Hotel
Season: Summer
Customer Amount: 3
Starting Date: 5.8.2022
Final Date: 10.8.2022
Day: 5
---Hotel Information---
Hotel Place: Mugla
Hotel Name: Liberty_Fabay
Hotel Type: full_pension
Hotel Price (Total): 15000



Shortcomings:

If project continues, we want to control quota of hotels and daily trips. We didn't have the chance to implement an algorithm that checks solidity ratio of hotel rooms or daily trip quotas.

We print information of reservation at the end of program but if user changes his/her mind, we do not obtain information that user wants to change again. We planned to ask user to whether s/he want to continue or not.

Conclusion:

We try to operate vacation reservation system. While running the system, we used object oriented programing. Moreover, we tried to use all the subjects that we have learned such as classes, inheritance, polymorphism, abstraction etc. and we tried to create an efficient program.