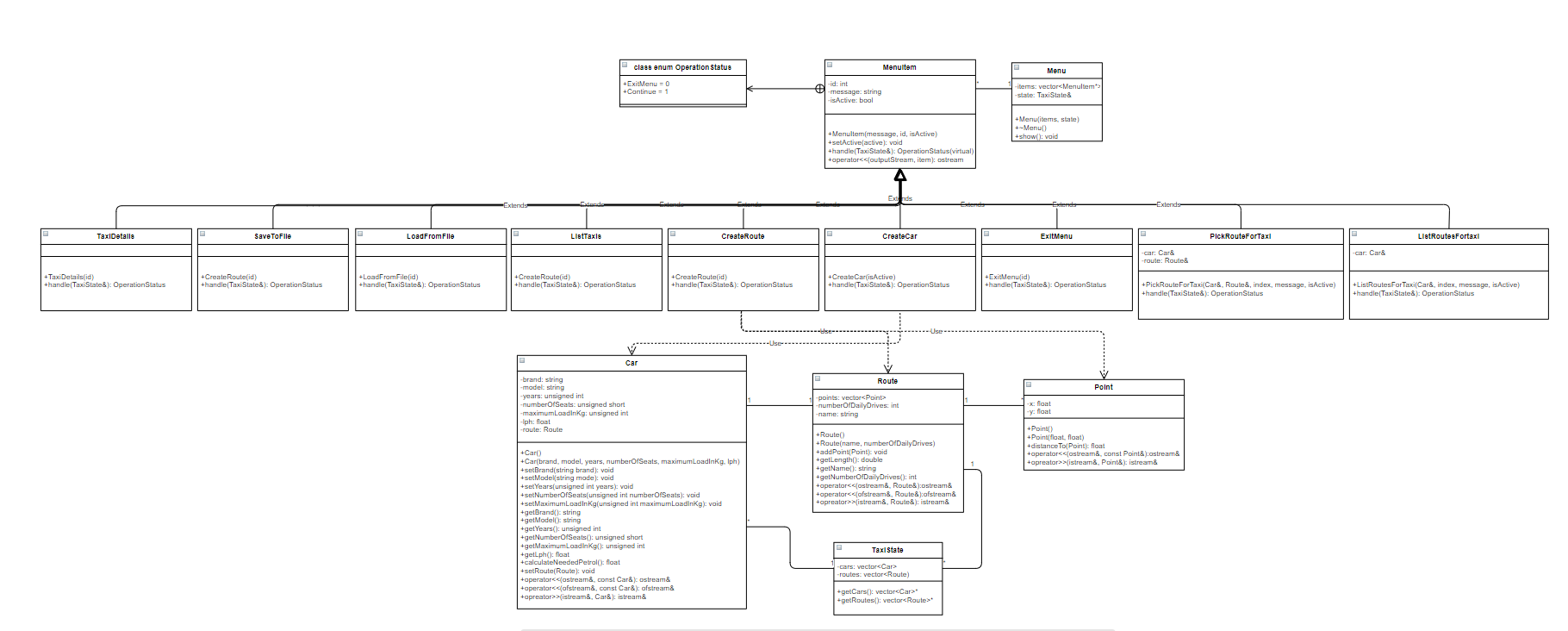
Курсов проект

по  
„Програмни езици“

|  |  |
| --- | --- |
| Изготвил: | Възложил: |
| Иван Ивайлов Милев | гл. ас. д-р Иван Станков |
|  |  |
| Дата: 01.2021. гр София |  |
| UML Class Диаграма |  |



# Описание на създадените функции/методи и класове:

## int main()

Начало на програмата – създава основното меню и вика методът show на менюто, за да го визуализира

## Class Menu

Обекти от тип Menu се грижат за визуализацията на меню и за неговата навигация. Всеки обект от тип Menu знае кои са неговите опции(MenuItem-и).

### void show()

При извикване методът ще визуализира опциите на менюто в командния ред, като се грижи за навигацията, която става посредством ‘w’ и ‘s’. При натискане на Enter дадена опция ще бъде избрана.

## Class Car

Репрезентация на такси. Всяко такси има марка, модел, брой места, на колко години е автомобила, максимална товароспособност, среден разход и маршрут.

### calculateNeededPetrol()

Пресмята колко литра гориво трябва на автомобила, за да изпълни маршрута нужния брой пъти(колкото е описано в самия маршрут)

## Class Route

Репрезентира маршрут, всеки маршрут има име и брой пъти, които се обикаля. Нужните getters и setters за тези член-променливи.

### double getLength()

Изчислява и връща дължината на маршрута.

## Class Point

Репрезентира точките от маршрутра, съдържа техните координати(географска дължина и ширина).

### float distanceTo(Point other)

изчислява разстоянието между текущата точка и подадената като аргумент

## Class TaxiState

Съдържа състоянието на програмата в два вектора един за колите и един за маршрутите.

## Class MenuItem

Всеки MenuItem има идентификатор(номер в менюто), съобщение(което да бъде показано в менюто) и дали е активна опция(избрана).

### void setActive(bool active)

Сетва дали е активен конкретния MenuItem.

### virtual OperationStatus handle(TaxiState& state)

Всеки MenuItem трябва да имплементира този метод, идеята на този метод е да бъде имплементиран от конкретен MenuItem, като този метод ще бъде извикан при избиране на опцията.

Всички последващи класове са наследници на MenuItem

## Class CreatCar: public MenuItem

Override-ва handle метода, като създава нова кола използвайки за параметри вход от потребителя чрез стандартния вход, новата кола е добавена към TaxiState

## Class CreateRoute: public MenuItem

Override-ва handle метода, като създава нов маршрут използвайки за параметри вход от потребителя чрез стандартния вход, новият маршрут е добавен към TaxiState

## Class ExitMenu: public MenuItem

Override-ва handle метода и връща ExitMenu, за да излезем от Menu-то

## Class PickRouteForTaxi: public MenuItem

Има две член променливи кола и маршрут.

Override-ва handle метода като, за съответната кола ще бъде добавен съответният маршрут като се изпише съобщение за колко литра гориво ще бъде нужно за пътуването.

## Class ListRoutesForTaxi: public MenuItem

Има член променлива – такси.

Override-ва handle метода, като той създава подменю от PickRouteForTaxi меню опции, ще има опция за избор на всеки маршрут от TaxiState.

## Class ListTaxis: public MenuItem

Override-ва handle метода, създава ново подменю което ще ипище всички таксита като опции(ListRoutesForTaxi).

## Class TaxiDetails: public MenuItem

Override-ва handle метода, като принтира на стандартния изход всикчи таксита и техните маршрути

## Class SaveToFile: public MenuItem

Override-ва handle метода като иска от потребителя да въведе име на файл, в който да се запази конфигурацията.

Запазват се елементите във векторите на TaxiState в текстови фаил като за записването се ипозлват override-и на операторите

std::ofstream& operator<<(std::ofstream& os, const Route& route)

std::ofstream& operator<<(std::ofstream& os, const Car& car)

## Class LoadFromFile: public MenuItem

Override-ва handle метода като иска от потребителя да въведе име на файл, от който да се прочете конфигурацията.

За четене от файл се използва – четене по ред и след това stringstream и оverride на операторите:

std::istream& operator>>(std::istream& is, Car& car)

std::istream& operator>>(std::istream& is, Route& route)  
std::istream& operator>>(std::istream& is, Point& point)

## Class Exit: public MenuItem

Override-ва handle метода като върща стойност за изход от менюто и така се напуска програмата.