

СЕМЕСТРИАЛНА ДОМАШНА РАБОТА

НА ГЕОРГИ МИЛЕНОВ СОКОЛОВ

НОМЕР: 21621397

СПЕЦИАЛНОСТ: КСТ

КУРС: I

ГРУПА: 4А

21621397 ГЕОРГИ МИЛЕНОВ СОКОЛОВ**Коли под наем**

Да се напише компютърна програма, реализираща информационна система за фирма, предоставяща коли под наем. Фирмата разполага с определен набор от коли (марка, модел, година на производство, регистрационен номер). При предоставяне под наем на кола, се съхранява информация за име и фамилия на клиент, ЕГН, възраст, категория правоспособност за шофиране, информация за заемана кола, ден от месеца, брой дни за заемане, статус: заета кола/върната кола. Максималният брой коли е 100.

Базова задача – сложност ниска

- A. Меню за избор на функциите в програмата. (7 седмица)
- B. Добавяне на коли. (7-8 седмица)
 - a. Добавяне на една нова кола.
 - b. Добавяне на списък с коли. Въвежда се цяло число n и след него n на брой коли.
- C. Извеждане на всички коли на екрана. (8-9 седмица)
 - a. Извеждане на коли с максимален брой дни на заемане
 - b. Извеждане на коли по определена марка

Допълнение първо – сложност средна (+ базова задача)

- D. Даване на кола под наем. (9-10 седмица)
 - a. Ако колата присъства в списъка и има налични бройки, както и ако данните на шофьора отговарят на изискванията (да има 18 години и съответна категория правоспособност за шофиране) се дава кола за необходимия период.
 - b. Ако няма налични бройки от дадения модел се предлага друг и при потвърждение от клиента, се дава новата кола.
 - c. Ако няма никакви свободни коли се извежда съответно съобщение

Допълнение второ – сложност висока (+ базова задача + допълнение първо)

- E. Връщане на кола. (10 -11 седмица)
 - a. Въвежда се регистрационния номер на колата и се маркира като върната.
 - b. Въвежда се име на клиента и се маркира като върната.

Допълнение трето – сложност висока (+ базова задача + допълнение първо + допълнение второ)

- F. Одит на колите в подменю. (11-12 седмица)
 - a. Извеждане на всички коли, заети за месеца, сортирани по регистрационен номер.
 - b. Извеждане на всички заемания на коли за даден клиент, сортирани по ден от месеца в намаляващ ред.
 - c. Извеждане на всички заемания (като бройка) за всички коли на фирмата, сортирани по брой заемания от най-много към най-малко.
- G. Данните в програмата да могат да се запазват във файл между две стартирания на програмата.

Допълнение четвърто – (за допълнителни точки)

- H. Допълнителни условия:
 - a. При добавянето на коли, не може да има дублиране с един и същ регистрационен номер.
 - b. При даването на кола под наем, на един клиент, може да се даде само една кола, за даден период от време.
 - c. Данните записани в двоичен файл да могат да се записват в текстов файл и обратно.

Описание на структурите и техните методи:

struct Calendar - помни текуща дата и съдържа функции за работа с дати. Съдържа методите:

- updateCalendar - за обновяване на текущата дата
- isDateReal - за проверка на валидност на дата
- isFebLonger - за проверка за високосна година
- isDateBigger - за сравнение на 2 дати
- calcDaysBetweenYears - брой дните между 2 год.
- calcDaysOngoingYear - брой дните от началото на дадена година
- calcDaysBetweenDates - брой дните между 2 дати
- doDateCheck - прави проверка на дата и я сравнява с настоящата

struct CarTable - помни важната информация на записаните коли. Съдържа методите:

- add - за добавяне на нова кола
- find - за намиране на индекса на търсена по рег. номер кола
- findFree - търси свободна кола по марка и модел или просто свободна (overloaded)
- changeState - сменя статуса на кола карана/некарана
- getLongestRental - връща най-дългото заемане на кола от таблицата

struct Client - помни важната информация за даден клиент. Съдържа полета и конструктор.

struct ClientList - списък от клиенти. Съдържа методите:

- add - добавя нов клиент
- find - намира клиент от списъка по лични данни
- findByName - намира кл. от списъка по 2 имена
- changeState - променя статуса на клиент (каращ/некаращ)

struct Rental - помни важната информация за едно наемане. Съдържа полета и конструктор.

struct RentalList - списък от наемания. Съдържа методите:

- add - добавя наемане
- returnRental - връща наемане
- find - намира последната комбинация от клиент и автомобилна регистрация
- findNotRet - намира последното наемане, което не е върнато по регистрация или клиент (overloaded)

struct Logic - отговаря за логически действия с всички гореизредени структури. Чрез нея се изпълняват основните функции на програмата. Съдържа методите:

- addCar - добавя кола или n коли (overloaded)
- addClient - добавя клиент в списъка с клиенти
- addRental - добавя наемане на кола в сп. с наемания
- loanCar - отдава кола от записаните под наем и ако трябва добавя и нов клиент
- returnCar - връща кола по наемане
- returnCarByReg - връща кола по рег. номер
- returnCarByNames - вр. кола по имена на клиент

- listSortedMaxRentalDays - връща подредени колите с най-дълго наемане
- listSortedBrand - връща подредени колите от зададена марка
- listSortedClientAndDate - връща всички наемания на даден кл. сорт. по дата

struct InputOutput - отговаря за входа и изхода на програмата. Съдържа методите:

- printMainMenu - изписва опциите за взаимодействие с програмата
- mainMenu - отговаря за избора на опция от менюто
- menuOne - 1^{во} подменю на mainMenu
- menuTwo - 2^{ро} подменю
- menuThree - 3^{то} подменю
- menuFour - 4^{то} подменю
- receiveChar - проверява дали входър е char
- receivePositive - проверява дали входът е пол. число

функция main - съдържа безкраен цикъл докато не се получи команда за прекратяване изпълнението на програмата.