Да се реализира приложение – опростена система на таксиметрова компания. Приложението трябва да може да поддържа 2 вида потребители – клиенти и шофьори. Те трябва да могат да се регистрират в системата и да влизат в нея чрез потребителско име и парола, както и да излизат от приложението, без то да приключва работа.

**За клиенти:**

* register <type> <username> <password>

<first\_name> <last\_name>

* login <username> <password>
* logout

**За шофьори:**

* register <type> <username> <password>

<first\_name> <last\_name>

<car\_number>

<phone\_number>

* login <username> <password>
* logout

Ако потребител влезе в системата като клиент, трябва да може да извършва следните действия:

* **order**

**<address>**

**<destination>**

Поръчва такси, като задава текущ адрес, дестинация и брой пътници.

Всеки адрес съдържа име, координати (точка с координати за x и y) и допълнителна уточняваща информация (не е задължителна за всеки адрес).

При изпращане на поръчка към приложението, системата трябва да прати известие до всички свободни шофьори, като първо праща на най-близкия до адреса, от който е поръчана. Когато някой шофьор приеме поръчката, системата уведомява потребителя. Системата генерира уникално id и го изписва на потребителя.

* **check\_order <id>**

Проверява състояние на поръчката – дали е приета, коя кола ще я изпълни, какви са данните на шофьора (име и фамилия, номер на автомобила, телефонен номер, рейтинг) и след колко време ще пристигне.

* **cancel\_order <id>**

Отменя поръчка. Ако някой шофьор вече е приел поръчката, системата го освобождава и го уведомява за това.

* **pay <id> <amount>**

Плаща за дадена поръчка, след като е отбелязана от шофьора като приключила. Парите, платени от потребителя, отиват в сметката на шофьора.

* **rate <name\_of\_driver> <rating>**

Да оцени (от 1 до 5) шофьор на поръчка. Поръчката, която оценява, трябва да е направена от него и вече да е завършена.

* **add\_money <amount>**

Добавя пари към сметката на потребителя. За удобство приемаме, че е въведена валидна кредитна или дебитна карта и не се правят допълнителни проверки за нея.

Ако потребител влезе в системата като шофьор, трябва да може да:

* **change\_address <address>**

Променя текущия си адрес (име и координати).

* **check\_messages**

Проверява съобщенията, изпратени от системата. Там са поръчките, които системата изпраща към него.

* **accept\_order <id> <minutes>**

Приема поръчка и изпраща към системата информация до колко минути ще е на адреса.

* **decline\_order <id>**

Отказва поръчка. Ако шофьор откаже поръчка, системата трябва да я предложи на най-следващия най-близък до адреса на поръчката шофьор.

* **finish\_order <id>**

Отбелязва поръчката като завършена (когато е стигнал до адреса). Това автоматично сменя адреса му на адреса на дестинацията.

* **accept\_payment <id> <amount>**

Приема плащане. Праща към системата информация колко пари е получил от нея.

Цялата информация за потребителите, пътуванията и общата печалба на таксиметровата фирма трябва да се съхраняват по подходящ начин във файлове.

**Примерна употреба на приложението:**

Какво вижда клиента на конзолата: Какво вижда шофьора на конзолата:

> register client client1 12345 > register driver driver1 54321

> Ivan Ivanov > Peter Petrov

> 1234

> 0881234567

> login client1 12345 > login driver1 54321

> order

> Address1 0 0 In front of the building

> Address2 5 5 > check\_messages

Order ID: 1 Ivan Ivanov has made an order from:

Address1 0 0 In front of the building

To:

Address2 5 5

> accept\_order Address1 Address2 20

Your driver Peter Petrov will arrive in 20

minutes!

> check\_order Address1 Address2

Driver: Peter Petrov

Car number: 1234

Phone number: 0881234567

> finish\_order Address1 Address2

> pay 10

> accept\_payment 10

За тестване ще бъде удобно, след като направите поръчка, да излезете от профила на клиента, да влезете от профила на шофьора, да приемете/откажете поръчката и отново да влезете от клиента и т.н.