

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ХАРКІВСЬКИЙ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ В.Н.КАРАЗІНА  
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК

Лабораторна робота №5  
з навчальної дисципліни  
Крос-платформне програмування

Виконав:  
студент групи КС-23  
Терещенко Є.Ю.

Перевірив:  
Канд. фіз.-мат. наук  
Споров О. Є.

Харків - 2024

## **Завдання №1.**

Напишіть простий розподілений клієнт / серверний додаток за допомогою TCP сокетів. У цьому додатку сервер приймає завдання від клієнтів, виконує ці завдання, визначає час їх виконання і потім повертає всю цю інформацію клієнту. При цьому саме клієнти створюють свої власні завдання і відправляють їх на сервер для виконання (клас завдання повинен реалізувати інтерфейс, визначений відповідно до договору з сервером). Визначення класу завдання відправляється клієнтом на сервер і, як тільки class - файл стає доступним, сервер може виконувати отримане завдання. Аналогічно, сервер створює об'єкт класу результату і відправляє його разом з визначенням класу клієнту. При цьому клас результату реалізує інтерфейс, відомий клієнту.

В якості клієнтського завдання можна взяти завдання обчислення факторіала досить великого числа.

## **Завдання №2**

Розглянемо взаємодію двох комп'ютерів в мережі. Перед тим, як почати роботу, комп'ютери повинні обмінятися IP-адресами. Це може стати досить непростим завданням. Створимо спеціальний UDP сервер, який допоможе комп'ютерів обмінятися «Координатами»: IP-адресами і номерами портів. Потім створимо UDP клієнтів, які перевіряють роботу сервера: відправляють запит, який реєструє відправив комп'ютер на сервері і отримують відповідь сервера - список вже зареєстрованих комп'ютерів

Для виконання даного завдання було створено два інтерфейси: Result (який реалізують класи, що повертають результат) та Executable (який реалізують класи, що виконують завдання, наприклад, обчислення факторіала).

У пакеті server розташовані класи, що відповідають за серверну частину, зокрема: Connection (клас для з'єднання), ResultImpl (реалізація інтерфейсу Result), ServerFrame (клас для роботи з сервером) та Start (клас для запуску сервера).

У пакеті client розміщені класи, що відповідають за клієнтську частину, зокрема: ClientFrame (клас для взаємодії з клієнтом) та Factorial (клас для обчислення факторіала).

Загалом, структура програми складається з інтерфейсів та класів, які реалізують ці інтерфейси, а також двох пакетів - server і client, які відповідають за серверну та клієнтську частини відповідно.

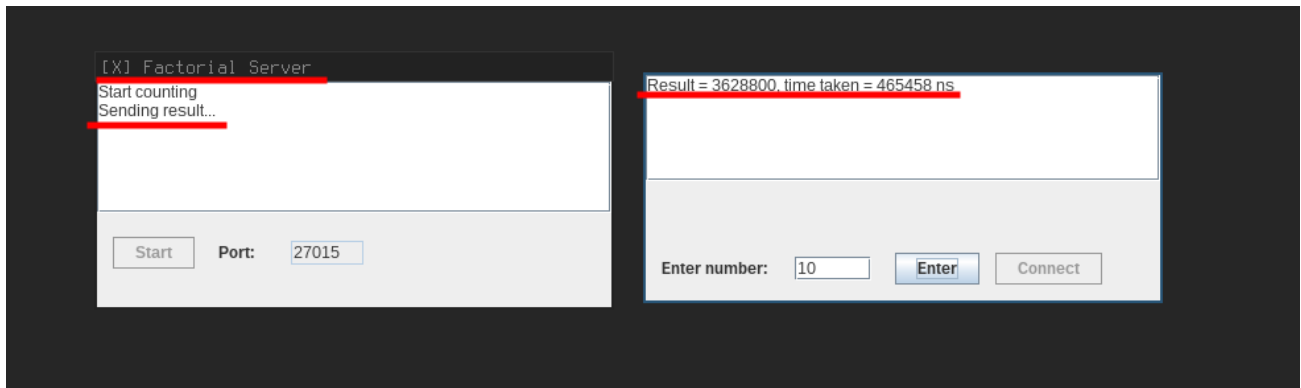


Рисунок 1 – Результат виконання першого завдання

Для вирішення цього завдання були створені наступні класи:

1. Клас User - цей клас призначений для зберігання інформації про Інтернет-адресу зареєстрованого комп'ютера. Крім того, він реалізує інтерфейс Serializable, що дозволяє об'єктам цього класу бути серіалізованими.
2. Клас ActiveUsers - цей клас використовується для зберігання інформації про всіх зареєстрованих комп'ютерів на даному сервері. Він містить методи та структури даних для управління списком активних користувачів.
3. Клас UDPServer - цей клас представляє сервер UDP. Він відповідає за обробку вхідних запитів та відправку відповідей через протокол UDP.
4. Клас UDPClient - цей клас представляє клієнта UDP. Він використовується для взаємодії з сервером UDP шляхом надсилання запитів та отримання відповідей через протокол UDP.

Server is up and running...

Waiting for new connections.New user has connected.  
Registering the user.  
User 127.0.0.1, port: 64344 has been registered.  
Sending a response...Response has been sent.

1

Waiting for new connections.New user has connected.  
Registering the user.  
User 127.0.0.1, port: 64344 is already in the database.  
Sending a response...Response has been sent.

3

Waiting for new connections.[]

Do you want to shut down the server after receiving data?

1. Yes

2. No.

2

Sending a request to the server... Request sent.

2

Requesting user list from the server...

User list received successfully.

Total online users: 1

List of active users:

Address: localhost, Port: 64344

~/IdeaProjects/rgr1 \$ **java** -classpath /home/kyryl/IdeaProjects/

Do you want to shut down the server after receiving data?

1. Yes

2. No.

2

Sending a request to the server... Request sent.

4

Requesting user list from the server...

User list received successfully.

Рисунок 2 – Результат виконання другого завдання

Посилання на GidHub: <https://github.com/zellii1/KPP.git>