Assignment Question

Design and implement a scalable client–server communication system using Python socket programming in which multiple server instances run concurrently on different port numbers within a single execution environment, and each server is capable of handling multiple clients simultaneously using multithreading.

The server application should:

- Run multiple servers on different port numbers in the same terminal.
- Handle multiple client connections concurrently for each server.
- Use multithreading to ensure that client requests are processed independently.
- Identify the type of task requested by the client and perform server-side processing accordingly.
- Continue running until manually terminated and handle client disconnections gracefully.

Server-Side Tasks

Each server must support the following two tasks based on the client request:

1. Arithmetic Processing Task

The client sends a request in the form of an arithmetic operation followed by two operands, such as:

```
add 10 20
sub 50 15
mul 6 7
div 40 5
```

The server must parse the request, perform the specified arithmetic operation, and return the computed result to the client.

2. String Analysis Task

The client sends a request in the form:

```
analyze Hello Server Programming
```

The server must analyze the given string by converting it to uppercase and determining the number of characters and words present in the string, and then return the analysis result to the client.