

國立臺南大學

資訊工程學系

網路程式設計

多執行序計算服務

**Multi-Thread Calculating Service**

學號：S10259002

姓名：黃詩豪

指導教授：楊文霖

中華民國 105 年 03 月 31 日

# 目錄

目錄.....	2
圖目表.....	3
表目錄.....	3
一、系統需求與目標（System Objectives） .....	4
二、系統主要功能描述（Description of requirement operations） .....	4
2-1、Server 端： .....	4
2-2、Client 端： .....	4
四、系統設計（System Design） .....	6
4-1、設計方法與工具.....	6
4-2、軟體組織架構.....	6
4-3、軟體元件模組設計 .....	7
五、系統說明與簡易使用者手冊（System User Guide） .....	8
5-1、Server 端執行畫面與操作說明 .....	8
5-2、Client 端執行畫面與操作說明 .....	9

## 圖目表

Fig. 1 Server-Client 溝通資料傳輸圖 .....	4
Fig. 2 Server-Client 計算服務系統流程圖 .....	5
Fig. 3 Multi-Thread 計算服務系統物件模式圖 .....	6
Fig. 7 Server 啟動畫面 .....	8
Fig. 8 接收 Client 資料與處理 .....	8
Fig. 9 多個 Client 請求 .....	9
Fig. 10 提示 Client 斷開連接 .....	9
Fig. 11 Client 啟動並提示輸入資料 .....	10
Fig. 12 請求的回傳結果 .....	10
Fig. 13 提示繼續請求 .....	10
Fig. 14 處理不合法的資料 .....	11

## 表目錄

Table. 1 MyServer 類別功能說明表格 .....	7
Table. 2 MyServerThread 類別功能說明表格 .....	7
Table. 3 MyClient 類別功能說明表格 .....	8

## 一、系統需求與目標 (System Objectives)

設計 Multi-Thread Server 與 Client 程式，在 Client 端，使用者以鍵盤輸入一連串數字，然後傳送至 Server 計算，Server 會找出此一連串數字中的最大值與最小值，最後再傳回 Client 端，而 Client 端的使用者可以不斷輸入一連串數字，直到使用者離開。

系統目標：

- 多執行序實作
- Server 計算最大與最小值並回傳
- Client 可重複請求直到離開

## 二、系統主要功能描述 (Description of requirement operations)

2-1、Server 端：

- 計算並回傳最大值與最小值
- 多執行序執行 Client 請求

2-2、Client 端：

- 輸入一連串數值傳送至 Server
- 可重複請求 Server 計算，直到使用者自願離開
- 例外處理功能

## 三、系統分析 (System Analysis)

系統主要分為 Server 與 Client 兩方，Server 需要計算從 Client 端傳送的一連串數字，然後找出其中的最大值與最小值，Client 端允許使用者重複請求 Server 計算最大值與最小值。但是當 Client 傳送資料給 Server 時，需要檢查使用者輸入的資料是否正確，如果不正確則提示使用者重新輸入直到正確，因此 Server 不會因為接收到的資料而出錯。

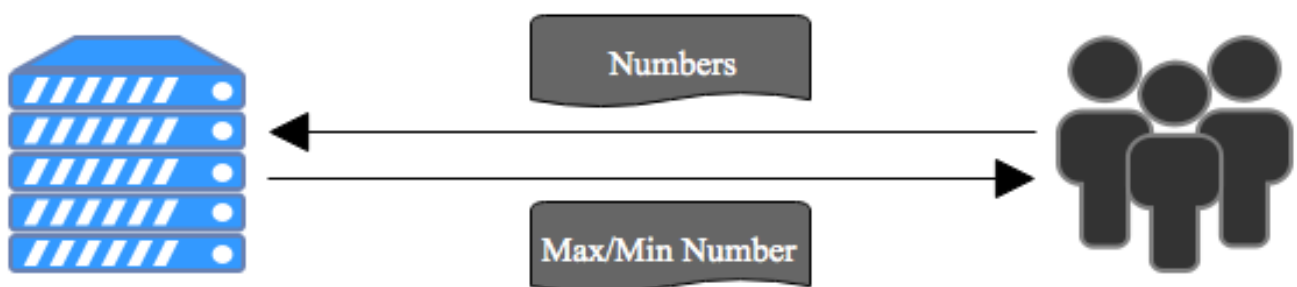


Fig. 1 Server-Client 溝通資料傳輸圖

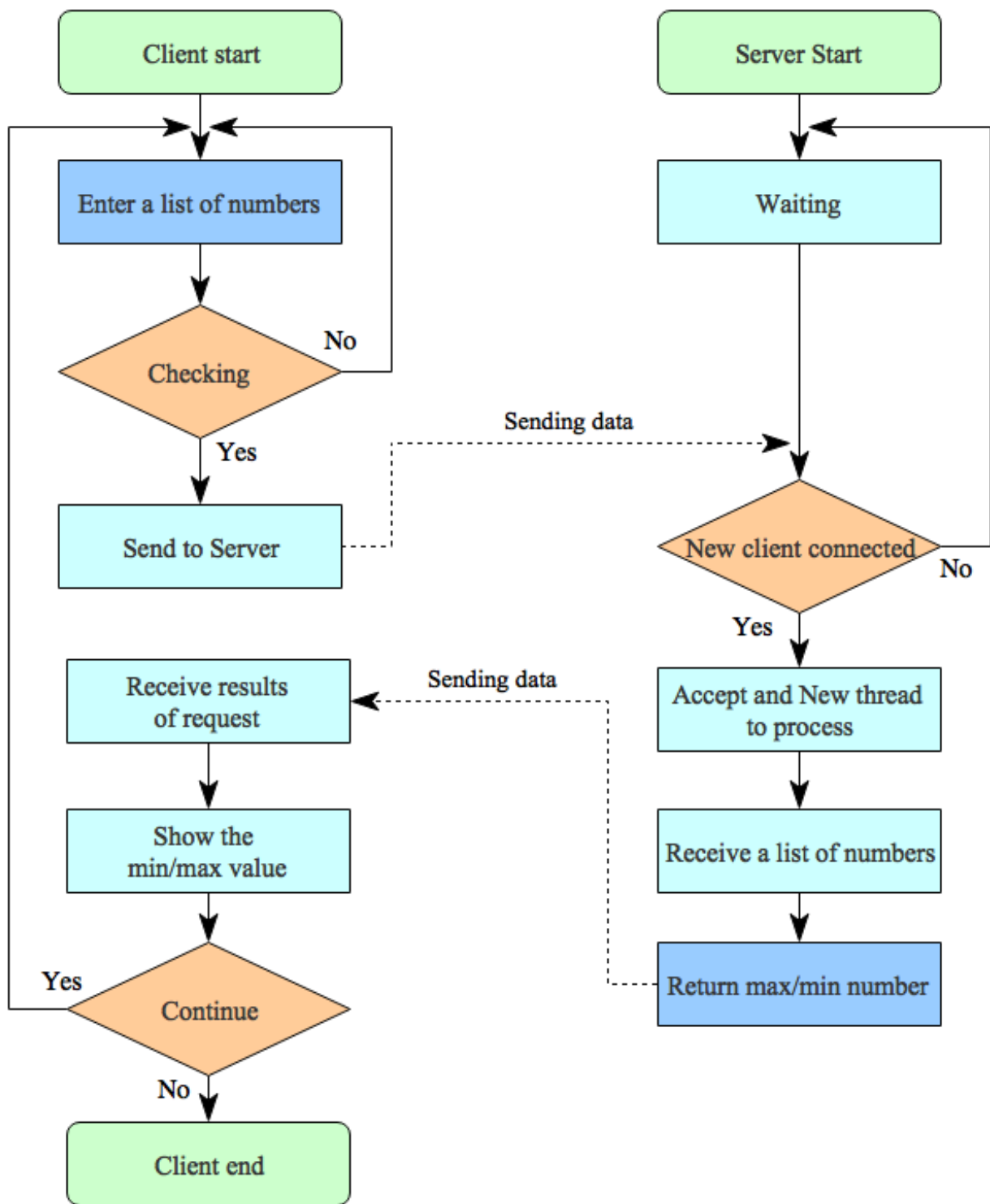


Fig. 2 Server-Client 計算服務系統流程圖

## 四、系統設計 (System Design)

### 4-1、設計方法與工具

Multi-Thread 計算服務使用物件導向設計，且是以 Server-Client 方式設計，系統分析工具則是使用 Word 與 Draw.io，程式開發工具為 Java 語言，在 Linux Mint 17.2 作業系統上使用 Eclipse MARS.2 開發，程式可於已安裝 JVM 之環境執行，介面設計則採用文字模式。

### 4-2、軟體組織架構

使用物件導向方式建立此系統，對於 Server 端，有 Server 主程式以及 Server Thread 程式，對於 Client 端，只有 Client 主程式。整個系統有三個類別，分別是 MyServer、MyServerThread 和 MyClient，MyServer 是伺服器端執行功能，當接收 Client 的連接時，產生一個 MyServerThread 物件，而此物件會計算 Client 傳送的一連串數字，並將此串數字中的最小值與最大值回傳。另外，MyClient 是客戶端執行程式，使用者將輸入一連串數字，然後會接收 Server 計算之結果並顯示，而當使用者選擇結束時，才將 Client 端的程式結束。整個系統的物件模式圖如。

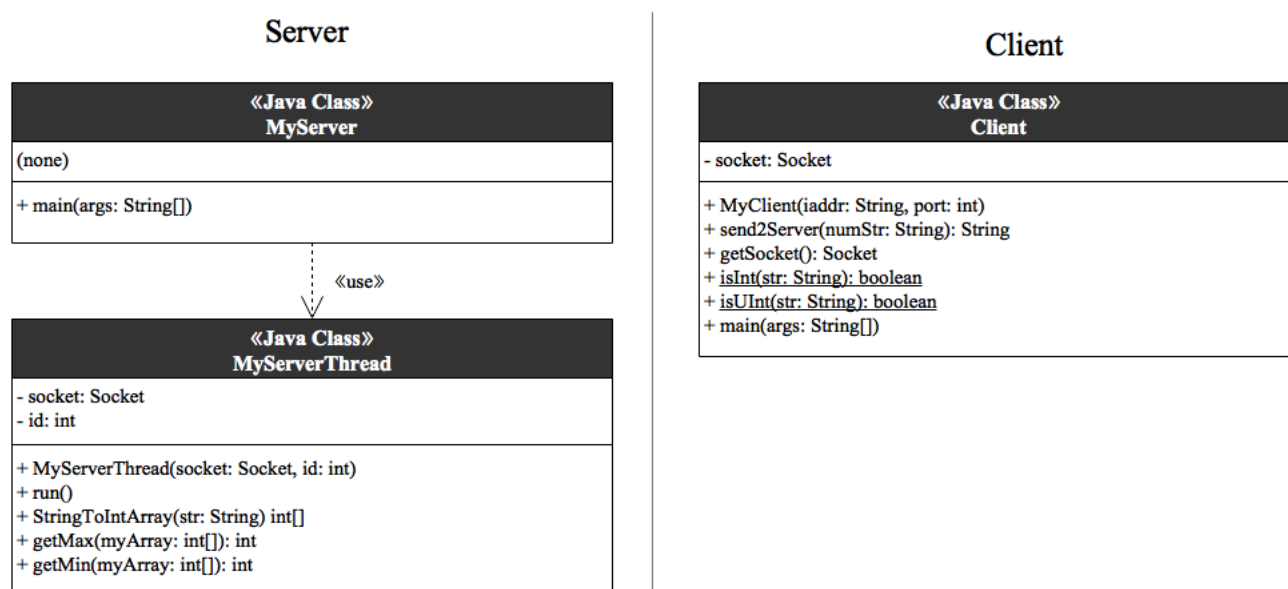


Fig. 3 Multi-Thread 計算服務系統物件模式圖

#### 4-3、軟體元件模組設計

- 伺服器類別 (MyServer Class)

功能名稱	功能說明
main(String[] args)	一直執行，當有 Client 連接時，接受 Client 的 Socket，然後建立新的執行序處理資料

Table. 1 MyServer 類別功能說明表格

- 伺服器執行序類別 (MyServerThread Class)

功能名稱	功能說明
MyServerThread(Socket socket, int id)	設定 Client 的 ID，以及接收 Client 的 Socket
run()	接收 Client 傳送的資料，並處理一連串數字，找出該串數字中最小值與最大值，然後回傳給 Client
StringToIntArray(String str)	將 Client 傳送的數字陣列字串轉成數字陣列
getMax(int[] myArray)	找出整數陣列中的最大值
getMin(int[] myArray)	找出整數陣列中的最小值

Table. 2 MyServerThread 類別功能說明表格

- 客戶端類別 (MyClient Class)

功能名稱	功能說明
MyClient(String iaddr, int port)	建立 Client 連接到 Server 的 Socket
send2Server(String numStr)	將整數數字陣列傳送至 Server，並回傳 Server 回應的結果
getSocket()	取得 Client 連接到 Server 的 Socket 物件
isInt(String str)	判斷此字串是否為整數
isUInt(String str)	判斷此字串是否為正整數
main(String[] args)	建立 MyClient 物件，然後提示使用者輸入一連串數字，接收到 Server 的回應後，詢問使用者要繼續使用此服務或是離開

Table. 3 MyClient 類別功能說明表格

## 五、系統說明與簡易使用者手冊 (System User Guide)

### 5-1、Server 端執行畫面與操作說明

- Server 啟動並等待 Client 連接  
等待 Client 連接。



Fig. 4 Server 啟動畫面

- 接收 Client 資料並處理與回傳  
接受 Client 連接，然後從陣列中找出最小值與最大值。



Fig. 5 接收 Client 資料與處理

- 能夠接受多個 Client 連接，並以多執行序運算



接受多個 Client 連接，並使用不同執行序處理 Client 的請求。

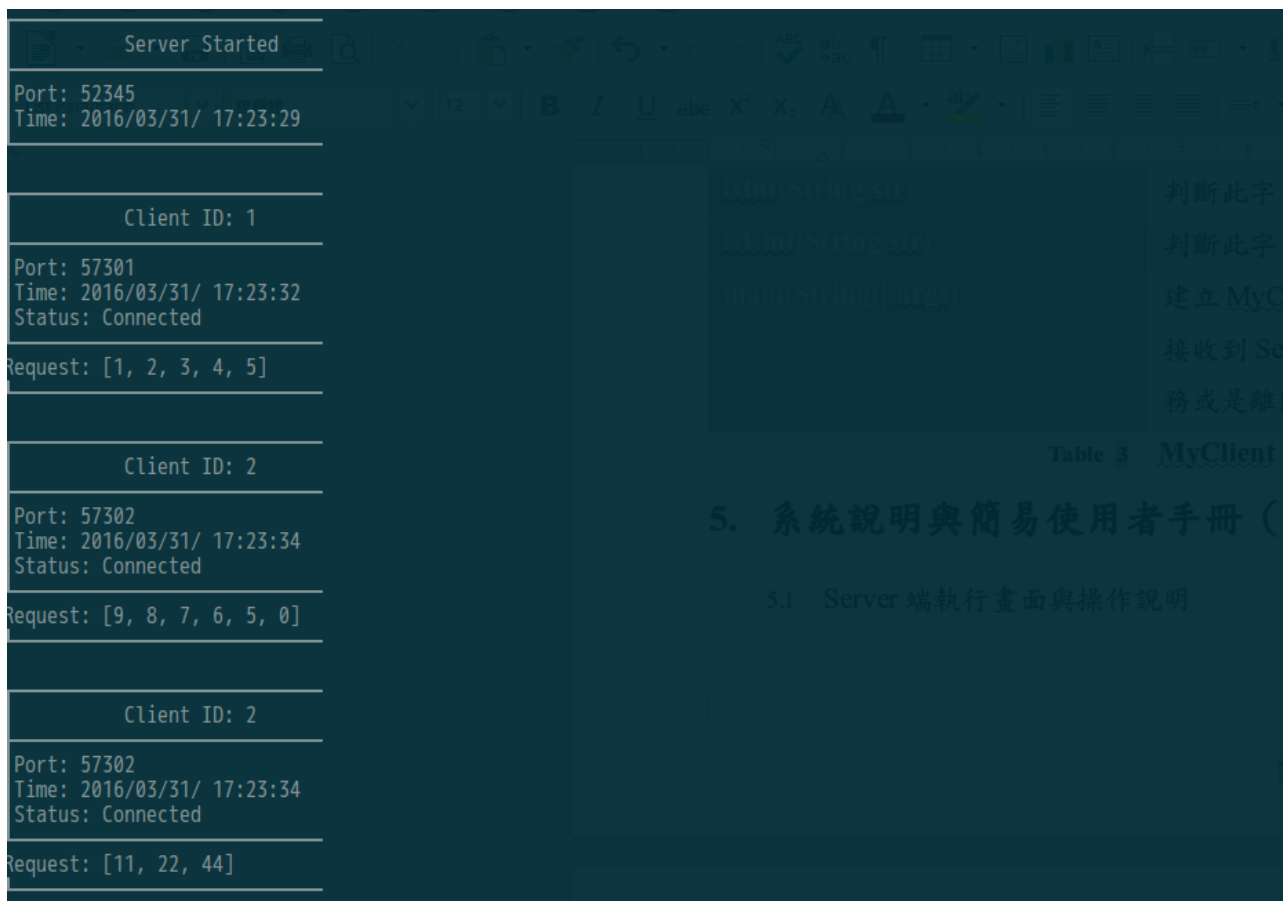


Fig. 6 多個 Client 請求

- Client 結束連接

當 Client 終止連接時，顯示相關資訊，以及是哪個 Client 結束連接。



Fig. 7 提示 Client 斷開連接

## 5-2、Client 端執行畫面與操作說明

- Client 啟動並提示使用者輸入陣列大小與數字

Client 當啟動時，提示使用者輸入陣列的大小，以及輸入所要計算最小值與最大值的一連串數字。



Fig. 8 Client 啟動並提示輸入資料

- 向 Server 請求，並接收回傳結果

輸入完一連串數字後，會傳送至 Server，待 Server 計算完成並回傳結果，將結果顯示於螢幕上。

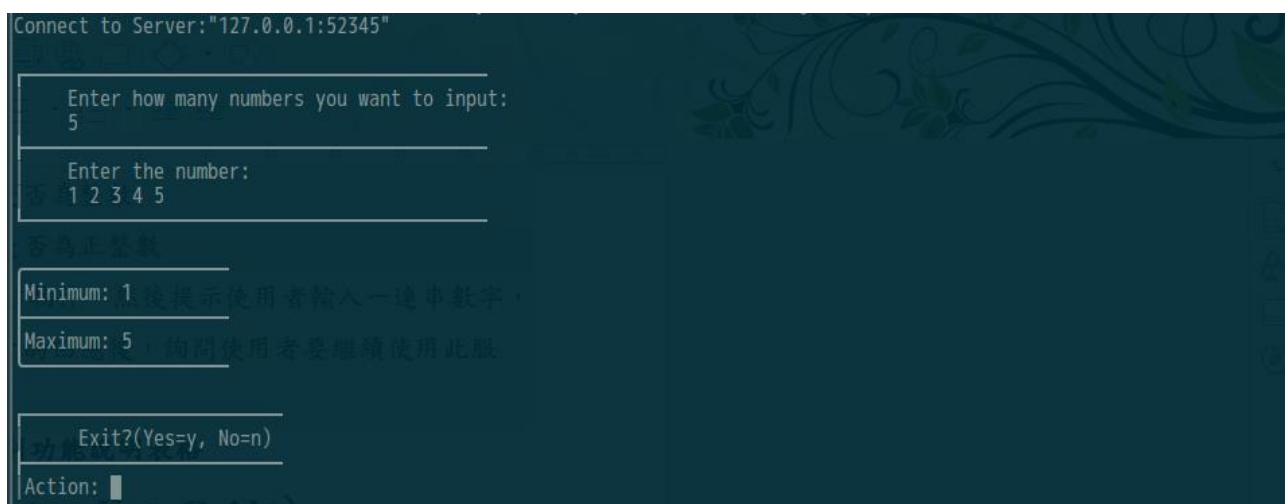


Fig. 9 請求的回傳結果

- 提示使用者是否繼續使用此服務

當完成一次的請求時，會提示使用者是否要繼續請求。



Fig. 10 提示繼續請求

- 例外處理，判斷輸入是否為數字

為了避免錯誤的資料傳送到 Server，以致 Server 發生錯誤而結束程序，因此在使用者輸入資料時，判斷輸入的資料是否符合整數值，當輸入的格式皆正確時，才將資料傳送至 Server。



```
Enter how many numbers you want to input:
y
Please enter positive integer:
█
```

Fig. 11 處理不合法的資料