

實驗三

- 1.1.1 client

- 目錄: 實驗三/client/udt4

- 執行環境: Linux

- 指令步驟:

1) 進入udt4/src/

2) make (編譯出libudt.so)

3) 設定libudt.so路徑(設定LD_LIBRARY_PATH)

4) 進入udt4/app/

5) make (編譯出執行檔udtclient)

6) 依照不同case選擇執行以下的script檔(run.sh=>20個clients, run50.sh=>50個clients, run80.sh=>80個clients, run100.sh=>100個clients) => 執行前需先依自己的環境改變script內指令的設定, 而指令的格式如下:

```
./udtclient <server_ip> <server_port> <output_interval(sec)>  
<mode(1 or 2)>
```

(p.s. <server_ip>: server的IP; <server_port>: server的port;

<output_interval(sec)>: 多久印一次資料, 單位是sec; <mode(1 or 2)>: mode1=>cumulative mode, 印出的結果為累積後的結果; mode2=>interval mode, 印出的結果為每個interval的結果)

E.g. ./udtclient 140.117.170.97 4455 300 1 &)

- 1.1.2 server

- 目錄: 實驗三/server/udt4/

- 執行環境: Linux

- 指令步驟:

1) 進入udt4/src/

2) make (編譯出libudt.so)

3) 設定libudt.so路徑(設定LD_LIBRARY_PATH)

4) 進入udt4/app/

5) make (編譯出執行檔udtserver)

```
./udtserver <server_port> <execute_time(sec)> <num_client>  
<output_interval(sec)> <tll(msec)>
```

(p.s. <server_port> : server的port; <execute_time(sec)>: 程式執行的時間;

<num_client> : client的個數; <output_interval(sec)>: 多久印一次資料;

<tll(msec)>: 設定tll, 單位是msec)