

## BÀI TẬP TUẦN 1 – LẬP TRÌNH MẠNG

### Cách thức nộp mã nguồn:

VS2015Project > HomeworkXX > TaskXXX				
Name	Date modified	Type	Size	
Debug	01-Feb-18 15:34	File folder		
ReadMe.txt	01-Feb-18 15:34	Text Document	2 KB	
stdafx.cpp	01-Feb-18 15:34	C++ Source File	1 KB	Chỉ nộp các file mã nguồn (.c, .cpp, .h)
stdafx.h	01-Feb-18 15:34	C Header File	1 KB	
targetver.h	01-Feb-18 15:34	C Header File	1 KB	
TaskXXX.cpp	01-Feb-18 15:34	C++ Source File	1 KB	
TaskXXX.vcxproj	01-Feb-18 15:34	VC++ Project	8 KB	
TaskXXX.vcxproj.filters	01-Feb-18 15:34	VC++ Project Filt...	2 KB	

Thay các ký tự X bằng chữ số thứ tự của bài tập như ví dụ dưới đây.

Đặt mã nguồn(.c, .cpp, .h) của mỗi Project vào thư mục riêng rẽ có tên thư mục là tên Project. Đóng gói các thư mục này vào file nén có tên theo định dạng HotenSV\_MSSV\_HW01.zip. Ví dụ với bài tập tuần này, cấu trúc file nén nộp như sau:

```
HotenSV_MSSV_HW01.zip
|-- UDP_Client
    |-- Các file mã nguồn(.c, .cpp, .h)
|-- UDP_Server
    |-- Các file mã nguồn(.c, .cpp, .h)
|-- TCP_Client
    |-- Các file mã nguồn(.c, .cpp, .h)
|-- TCP_Server
    |-- Các file mã nguồn(.c, .cpp, .h)
```

***Sinh viên nộp không đúng theo hướng dẫn sẽ không được chấm bài  
Gian lận khi làm bài tập sẽ nhận điểm F cho môn học này.***

**Bài 1.** Viết ứng dụng phân giải tên miền sử dụng UDP socket:

#### Server:

- Chạy ở số hiệu cổng bất kỳ theo tham số dòng lệnh

Cú pháp: **UDP\_Server.exe PortNumber**

Ví dụ: **UDP\_Server.exe 5500**

- Nhận yêu cầu phân giải tên miền hoặc địa chỉ IP cho client gửi lên
- Trả lại kết quả đáp ứng yêu cầu phân giải
- Yêu cầu bắt buộc: **Chương trình server không được tự động kết thúc trong mọi tình huống xử lý.**

#### Client:

- Kết nối tới server. Sử dụng tham số dòng lệnh cho địa chỉ IP và số hiệu cổng của server sẽ gửi yêu cầu tới.

Cú pháp: `UDP_Client.exe ServerIPAddress ServerPortNumber`

Ví dụ: `UDP_Client.exe 127.0.0.1 5500`

- Người dùng nhập tên miền từ bàn phím
- Client gửi yêu cầu tới server
- Nhận kết quả từ server và hiển thị
- Chức năng lặp lại cho tới khi người dùng nhập vào một xâu rỗng. Minh họa kết quả tại client:

INPUT	OUTPUT
google.com	IP Addresses: 216.58.197.110 216.58.197.123 126.58.99.199
aznsc.test.com	Not found information

### Thang điểm:

- Thực hiện đúng chức năng phân giải thuận: 5 điểm
- Thiết kế và sử dụng thông điệp có prefix: 2 điểm
- Tổ chức và trình bày mã nguồn: 3 điểm
- Một số lỗi sau đây bị trừ điểm:
  - Không liệt kê đủ địa chỉ IP khi phân giải thuận: -2 điểm
  - Không thực hiện được phân giải với 1 dạng giá trị nào đó: -1 điểm
  - Lỗi runtime error khiến client kết thúc: -1 điểm
  - Lỗi runtime error khiến server kết thúc: -3 điểm
  - Server không phục vụ được liên tục cho nhiều client: -2 điểm
  - Các lỗi khác: trừ điểm tùy theo mức độ nghiêm trọng của lỗi

### Yêu cầu môi trường:

- Công cụ phát triển ứng dụng: Microsoft Visual Studio 2015 Community
- Tên solution: Homework01
- Tên 2 project: UDP\_Server và UDP\_Client

### Bài 2. Viết ứng dụng xử lý xâu sử dụng TCP Socket:

#### Server:

- Chạy ở số hiệu cổng bất kỳ theo tham số dòng lệnh

Cú pháp: `TCP_Server.exe PortNumber`

Ví dụ: `TCP_Server.exe 5500`

- Nhận một xâu do client gửi lên. Nếu xâu chứa ký tự khác chữ số thì báo lỗi. Ngược lại, tính tổng các chữ số trong xâu và gửi kết quả cho client.
- Yêu cầu bắt buộc: **Chương trình server không được tự động kết thúc trong mọi tình huống xử lý.**

**Client:**

- Kết nối tới server. Sử dụng tham số dòng lệnh cho địa chỉ IP và số hiệu cổng của server sẽ gửi yêu cầu tới.

Cú pháp: **TCP\_Client.exe ServerIPAddress ServerPortNumber**

Ví dụ: **TCP\_Client.exe 127.0.0.1 5500**

- Cho phép người dùng nhập vào từ bàn phím một xâu
- Client gửi xâu mà người dùng nhập từ bàn phím tới server
- Nhận kết quả từ server và hiển thị
- Chức năng lặp lại cho tới khi người dùng nhập vào một xâu rỗng. Minh họa kết quả tại client:

INPUT	OUTPUT
123	6
123abc	Failed: String contains non-number character.
123@	Failed: String contains non-number character.

**Yêu cầu khác: Quá trình truyền tin phải xử lý vấn đề truyền dòng****Yêu cầu môi trường:**

- Công cụ phát triển ứng dụng: Microsoft Visual Studio 2015 Community
- Tên solution: Homework01
- Tên 2 project: TCP\_Server và TCP\_Client

**Thang điểm:**

- Thực hiện đúng chức năng: 5 điểm
- Xử lý được truyền dòng: 2 điểm
- Tổ chức và trình bày mã nguồn: 3 điểm
- Một số lỗi sau đây bị trừ điểm:
  - Lỗi runtime error khiến client kết thúc: -1 điểm
  - Các tình huống khiến server kết thúc: -3 điểm
  - Các lỗi khác: trừ điểm tùy theo mức độ nghiêm trọng của lỗi

**Gợi ý:****1. Thiết kế thông điệp trả lời từ server**

Kết quả trả về từ sever có 2 loại:

- Thông điệp chứa kết quả
- Thông điệp báo lỗi.

Cần thiết kế để phân biệt 2 loại thông điệp này bằng cách sử dụng prefix (Ký tự/Nhóm ký tự bắt đầu thông điệp) khác nhau. Client dựa trên prefix để xác định thông điệp có chứa kết quả dạng nào.

Một gợi ý thiết kế như sau:

- Prefix là '+' để báo thành công. Ví dụ: +202.191.56.65
- Prefix là '-' để báo thất bại. Ví dụ: -Not found information

## 2. Sử dụng tham số dòng lệnh

<https://docs.microsoft.com/en-us/cpp/c-language/parsing-c-command-line-arguments?view=msvc-160>

## 3. Xử lý vấn đề truyền dòng

Xem lại bài giảng trên lớp. Để kiểm tra vấn đề xử lý truyền dòng có thể sử dụng các chương trình cung cấp:

- Sử dụng Test1.cpp nếu phương pháp xử lý là dùng ký tự báo kết thúc thông điệp. Sửa các giá trị hằng số sau cho phù hợp:
  - SERVER\_PORT: Số hiệu cổng dịch vụ mà server sử dụng
  - SERVER\_ADDR: Địa chỉ IP của server
  - ENDING\_DELIMITER: Ký tự báo kết thúc thông điệp
- Sử dụng Test2.cpp nếu phương pháp xử lý là truyền kích thước (4 byte) theo thông điệp. Sửa các giá trị hằng số sau cho phù hợp:
  - SERVER\_PORT: Số hiệu cổng dịch vụ mà server sử dụng
  - SERVER\_ADDR: Địa chỉ IP của server

Nếu hiển thị thông báo "Stream test fail!" tức là vấn đề truyền dòng chưa được xử lý đúng.

## 4. Các vấn đề về tổ chức, trình bày mã nguồn

- Clean code: <https://viblo.asia/p/tom-tat-cuon-clean-code-cua-uncle-bob-6J3Zg07MImB>
- Xây dựng chương trình thành các hàm mô-đun chức năng
- Sử dụng quy ước định danh. Ví dụ:
  - Camel case: [https://en.wikipedia.org/wiki/Camel\\_case](https://en.wikipedia.org/wiki/Camel_case)
  - Snake case: [https://en.wikipedia.org/wiki/Snake\\_case](https://en.wikipedia.org/wiki/Snake_case)
  - Quy ước khác: <https://viblo.asia/p/naming-rules-cac-quy-tac-vang-trong-lang-dat-ten-ByEZkMXE5Q0>
- Comment cho các hàm mô-đun theo quy tắc thống nhất. Ví dụ:

```

/**
 * @function splitString: Split a string into one alphabetic string and one numeric string.
 *
 * @param source: A pointer to a input string.
 * @param letters: A pointer to a string contains all of letters in source string.
 * @param numbers: A pointer to a string contains all of numbers in source string.
 *
 * @return: 0 if success.
 *         1 if an error occurs because source string has any special character.
 */
int splitString(char *source, char *letters, char *numbers);

```

## 5. Kiểm thử

Sinh viên có thể tự thực hiện kiểm thử theo kịch bản sau(không bắt buộc tuân theo):

Bước	Cửa sổ Command Prompt 1	Cửa sổ Command Prompt 2	Cửa sổ Command Prompt 3	Cửa sổ Command Prompt 4
1	Khởi động server			
2		Khởi động client 1		
3			Khởi động client 2	
4				Khởi động client 3
5				Nhập 1 xâu nào đó
6			Nhập 1 xâu nào đó	
7		Nhập các xâu kiểm thử tùy ý		
8		Nhập xâu rỗng		
9			Nhấn Ctrl+C	
10				Nhập các xâu kiểm thử tùy ý
12		Khởi động client 1		
13		Nhập 1 xâu nào đó		
14		Nhấn Ctrl+C		
15				Nhập các xâu kiểm thử tùy ý
16				Nhập xâu rỗng