

BỘ ĐỀ ÔN THI LẬP TRÌNH MẠNG – PHẦN UDP
(Phần này sẽ lọc theo các MSV từ 001 đến hết. Cập nhật lần cuối: 01/12/2024)

MỤC LỤC

BÀI 1. TÌM SỐ CÒN THIẾU[002]	1
BÀI 2. ĐẾM SỐ LẦN XUẤT HIỆN[002]	1
BÀI 3. LỌC KÝ TỰ [010]	2
BÀI 4. MÃ HOÁ CAESAR [014]	2
BÀI 5. KÝ TỰ XUẤT HIỆN NHIỀU LẦN NHẤT[011].....	3
BÀI 6. KHÁCH HÀNG [008]	3
BÀI 7. BÀI TOÁN QUẢN LÝ SINH VIÊN [001].....	4
BÀI 8. CHUẨN HOÁ XÂU KÝ TỰ [001]	4
BÀI 9. HIỆU CỦA HAI TẬP KÝ TỰ[020].....	5
BÀI 10. TỔNG CÁC CHỮ SỐ [001]	5
BÀI 11. LIỆT KÊ SỐ NGUYÊN TỐ [B17DCAT176]	5

BÀI 1. TÌM SỐ CÒN THIẾU[002]

[Mã câu hỏi (qCode): wAKCZwj]. Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2207. Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản:

- a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;73457A17”
- b. Nhận thông điệp là một chuỗi từ server theo định dạng “requestId;n;A1,A2,...An” , với
 - requestId là chuỗi ngẫu nhiên duy nhất
 - n là một số ngẫu nhiên nhỏ hơn 100.
 - A1, A2 ... Am ($m \leq n$) là các giá trị ngẫu nhiên nhỏ hơn hoặc bằng n và có thể trùng nhau.

Ex: requestId;10;2,3,5,6,5

- c. Tìm kiếm các giá trị còn thiếu và gửi lên server theo định dạng “requestId;B1,B2,...,Bm”

Ex: requestId;1,4,7,8,9,10

- d. Đóng socket và kết thúc chương trình.

BÀI 2. ĐẾM SỐ LẦN XUẤT HIỆN[002]

[Mã câu hỏi (qCode): vSgxl3HQ]. Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2208. Yêu cầu xây dựng chương trình client trao đổi thông tin với server theo kịch bản:

- a. Gửi một thông điệp chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;9F8C2D3A”.

- b. Nhận một thông điệp từ server theo định dạng "requestId;data", với:

requestId là chuỗi ngẫu nhiên duy nhất.

data là một chuỗi ký tự liên tiếp cần xử lý. Ví dụ: "requestId;aaabbbccdaa"

c. Xử lý chuỗi bằng cách đếm số lượng ký tự và gom chúng lại theo định dạng "số_lần_ký_tự".
Gửi kết quả về server theo định dạng:

"requestId;processedData"

Ví dụ: Với chuỗi "aaabbbccdaa", kết quả sẽ là "requestId;5a3b2c1d"

d. Đóng socket và kết thúc chương trình.



Dựa theo log xử lý, đề bài nên sửa là: Đếm số lần xuất hiện các ký tự trong cả xâu, liệt kê theo thứ tự xuất hiện

BÀI 3. LỌC KÝ TỰ [010]

[Mã câu hỏi (qCode): GfeNSBMT]. [Loại bỏ ký tự đặc biệt, số, trùng và giữ nguyên thứ tự xuất hiện]

Một chương trình server cho phép kết nối qua giao thức UDP tại cổng 2208 . Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản dưới đây:

- Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng ";studentCode;qCode". Ví dụ: ";B15DCCN001;06D6800D"
- Nhận thông điệp là một chuỗi từ server theo định dạng "requestId;strInput"
 - requestId là chuỗi ngẫu nhiên duy nhất
 - strInput là chuỗi thông điệp cần xử lý
- Thực hiện loại bỏ ký tự đặc biệt, số, ký tự trùng và giữ nguyên thứ tự xuất hiện của chúng. Gửi thông điệp lên server theo định dạng "requestId;strOutput", trong đó strOutput là chuỗi đã được xử lý ở trên
- Đóng socket và kết thúc chương trình.

BÀI 4. MÃ HOÁ CAESAR [014]

[Mã câu hỏi (qCode): J5SE2YXc]. Mật mã caesar, còn gọi là mật mã dịch chuyển, để giải mã thì mỗi ký tự nhận được sẽ được thay thế bằng một ký tự cách nó một đoạn s. Ví dụ: với s = 3 thì ký tự "A" sẽ được thay thế bằng ký tự "D".

Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2207. Yêu cầu xây dựng chương trình client trao đổi thông tin với server theo kịch bản mô tả dưới đây:

- Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng ";studentCode;qCode". Ví dụ: ";B15DCCN001;825EE3A7"
- Nhận thông điệp là một chuỗi từ server theo định dạng "requestId;strEncode;s".
 - requestId là chuỗi ngẫu nhiên duy nhất
 - strEncode là chuỗi thông điệp bị mã hóa
 - s là số nguyên chứa giá trị độ dịch của mã
- Giải mã tìm thông điệp ban đầu và gửi lên server theo định dạng "requestId;strDecode"
- Đóng socket và kết thúc chương trình.

BÀI 5. KÝ TỰ XUẤT HIỆN NHIỀU LẦN NHẤT[011]

[Mã câu hỏi (qCode): CvlqJmaa]. Một chương trình server cho phép kết nối qua giao thức UDP tại cổng 2208. Yêu cầu là xây dựng một chương trình client tương tác với server kịch bản dưới đây:

a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;EE29C059”

b. Nhận thông điệp từ server theo định dạng “requestId; data”

- requestId là một chuỗi ngẫu nhiên duy nhất

- data là chuỗi dữ liệu đầu vào cần xử lý

Ex: “requestId;Qnc8d5x78aldSGWWmaAAjyg3”

c. Tìm kiếm ký tự xuất hiện nhiều nhất trong chuỗi và gửi lên server theo định dạng “requestId;ký tự xuất hiện nhiều nhất: các vị trí xuất hiện ký tự đó”

ví dụ: “requestId;8:4,9,”

d. Đóng socket và kết thúc chương trình

BÀI 6. KHÁCH HÀNG [008]

[Mã câu hỏi (qCode): 4MpxJ947]. Thông tin khách hàng được yêu cầu thay đổi định dạng lại cho phù hợp với khu vực, cụ thể:

a. Tên khách hàng cần được chuẩn hóa theo định dạng mới. Ví dụ: nguyen van hai duong -> DUONG, Nguyen Van Hai

b. Ngày sinh của khách hàng đang ở dạng mm-dd-yyyy, cần được chuyển thành định dạng dd/mm/yyyy. Ví dụ: 10-11-2012 → 11/10/2012

c. Tài khoản khách hàng được tạo từ các chữ cái in thường được sinh tự động từ họ tên khách hàng. Ví dụ: nguyen van hai duong → nvhduong

Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2209. Yêu cầu là xây dựng một chương trình client giao tiếp với server theo mô tả sau:

a. Đối tượng trao đổi là thể hiện của lớp UDP.Customer được mô tả như sau

- Tên đầy đủ của lớp: UDP.Customer

- Các thuộc tính: id String, code String, name String, , dayOfBirth String, userName String

- Một Hàm khởi tạo với đầy đủ các thuộc tính được liệt kê ở trên

- Trường dữ liệu: private static final long serialVersionUID = 20151107;

b. Client giao tiếp với server theo các bước

- Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;EE29C059”

- Nhận thông điệp chứa: 08 byte đầu chứa chuỗi requestId, các byte còn lại chứa một đối tượng là thể hiện của lớp Customer từ server. Trong đó, các thuộc tính id, code, name, dayOfBirth đã được thiết lập sẵn.

- Yêu cầu thay đổi thông tin các thuộc tính như yêu cầu ở trên và gửi lại đối tượng khách hàng đã được sửa đổi lên server với cấu trúc:

08 byte đầu chứa chuỗi requestId và các byte còn lại chứa đối tượng Customer đã được sửa đổi.

- Đóng socket và kết thúc chương trình.

BÀI 7. BÀI TOÁN QUẢN LÝ SINH VIÊN [001]

[Mã câu hỏi (qCode): QABGDFxd]. Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2209. Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản sau:

Đối tượng trao đổi là thể hiện của lớp UDP.Student được mô tả:

- Tên đầy đủ lớp: UDP.Student
- Các thuộc tính: id String, code String, name String, email String
- 02 Hàm khởi tạo:
 - o public Student(String id, String code, String name, String email)
 - o public Student(String code)
- Trường dữ liệu: private static final long serialVersionUID = 20171107

Thực hiện:

- Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;EE29C059”
- b. Nhận thông điệp chứa: 08 byte đầu chứa chuỗi requestId, các byte còn lại chứa một đối tượng là thể hiện của lớp Student từ server. Trong đó, các thông tin được thiết lập gồm id và name.
- c. Yêu cầu:
 - Chuẩn hóa tên theo quy tắc: Chữ cái đầu tiên in hoa, các chữ cái còn lại in thường và gán lại thuộc tính name của đối tượng
 - Tạo email ptit.edu.vn từ tên người dùng bằng cách lấy tên và các chữ cái bắt đầu của họ và tên đệm. Ví dụ: nguyen van tuan nam → namnvt@ptit.edu.vn. Gán giá trị này cho thuộc tính email của đối tượng nhận được
 - Gửi thông điệp chứa đối tượng xử lý ở bước c lên Server với cấu trúc: 08 byte đầu chứa chuỗi requestId và các byte còn lại chứa đối tượng Customer đã được sửa đổi.
- d. Đóng socket và kết thúc chương trình.

BÀI 8. CHUẨN HOÁ XÂU KÝ TỰ [001]

[Mã câu hỏi (qCode): NRVwBVvx]. Một chương trình server cho phép kết nối qua giao thức UDP tại cổng 2208. Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản dưới đây:

- a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng “;studentCode;qCode”. Ví dụ: “;B15DCCN001;5B35BCC1”
- b. Nhận thông điệp từ server theo định dạng “requestId;data”
 - requestId là một chuỗi ngẫu nhiên duy nhất
 - data là chuỗi dữ liệu cần xử lý
- c. Xử lý chuẩn hóa chuỗi đã nhận thành theo nguyên tắc
 - i. Ký tự đầu tiên của từng từ trong chuỗi là in hoa
 - ii. Các ký tự còn lại của chuỗi là in thường

Gửi thông điệp chứa chuỗi đã được chuẩn hóa lên server theo định dạng "requestId;data"

d. Đóng socket và kết thúc chương trình

BÀI 9. HIỆU CỦA HAI TẬP KÝ TỰ [020]

[Mã câu hỏi (qCode): JQCO3izC]. [Loại bỏ ký tự đặc biệt và ký tự trùng giữ nguyên thứ tự xuất hiện]

Một chương trình server cho phép kết nối qua giao thức UDP tại cổng 2208. Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản dưới đây:

- a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng ";studentCode;qCode". Ví dụ: ";B15DCCN001;B34D51E0"
- b. Nhận thông điệp là một chuỗi từ server theo định dạng "requestId;str1;str2".
 - requestId là chuỗi ngẫu nhiên duy nhất
 - str1, str2 lần lượt là chuỗi thứ nhất và chuỗi thứ hai
- c. Loại bỏ các ký tự trong chuỗi thứ nhất mà xuất hiện trong chuỗi thứ hai, giữ nguyên thứ tự xuất hiện. Gửi thông điệp là một chuỗi lên server theo định dạng "requestId;strOutput", trong đó chuỗi strOutput là chuỗi đã được xử lý ở trên.
- d. Đóng socket và kết thúc chương trình.

BÀI 10. TỔNG CÁC CHỮ SỐ [001]

[Mã câu hỏi (qCode): 0Iend7Pp]. Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2207. Yêu cầu là xây dựng chương trình client trao đổi thông tin với server theo kịch bản:

- a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng ";studentCode;qCode". Ví dụ: ";B15DCCN011;A1F3D5"
- b. Nhận thông điệp là một chuỗi từ server theo định dạng "requestId;num", với:
 - requestId là chuỗi ngẫu nhiên duy nhất.
 - num là một số nguyên lớn.
- c. Tính tổng các chữ số trong num và gửi lại tổng này về server theo định dạng "requestId;sumDigits".
- d. Đóng socket và kết thúc chương trình.

BÀI 11. LIỆT KÊ SỐ NGUYÊN TỐ [B17DCAT176]

[Mã câu hỏi (qCode): 78CCQ6xD]. Một chương trình server cho phép giao tiếp qua giao thức UDP tại cổng 2207. Yêu cầu là xây dựng một chương trình client trao đổi thông tin với server theo kịch bản:

- a. Gửi thông điệp là một chuỗi chứa mã sinh viên và mã câu hỏi theo định dạng ";studentCode;qCode". Ví dụ: ";B15DCCN009;F3E8B2D4".
- b. Nhận thông điệp là một chuỗi từ server theo định dạng "requestId;n, n", với:
 - requestId là chuỗi ngẫu nhiên duy nhất.
 - n là một số nguyên ngẫu nhiên ≤ 100 .

- c. Tính và gửi về server danh sách n số nguyên tố đầu tiên theo định dạng "requestId;p1,p2,...,pk", trong đó $p1, p2, \dots, pk$ là các số nguyên tố.
- d. Đóng socket và kết thúc chương trình.