

Bài tập 01.01 – Đóng gói dữ liệu

- Ứng dụng **info_client** cho phép người dùng nhập tên máy tính (là chuỗi ký tự), danh sách các ổ đĩa (gồm ký tự và kích thước ổ đĩa) từ bàn phím. Các dữ liệu này sau đó được đóng gói và chuyển sang **info_server**.
- Ứng dụng **info_server** nhận dữ liệu từ **info_client**, tách các dữ liệu và in ra màn hình.

Ví dụ:

+ Tên máy tính MY_LAPTOP_DELL

+ Số ổ đĩa 3

C – 500GB

D – 250GB

E – 250GB



Bài tập 01.02 – Xử lý dữ liệu streaming

- Ứng dụng **client** đọc nội dung file văn bản và gửi sang **server**.
- Ứng dụng server nhận dữ liệu từ client, in ra màn hình số lần xuất hiện chuỗi ký tự (server không cần tạo file để chứa nội dung nhận được): “0123456789”
- Chú ý cần xử lý trường hợp khi chuỗi “0123456789” nằm giữa 2 lần truyền.

Ví dụ nội dung file văn bản:

SOICTSOICT01234567890123456789012345678901234567
89SOICTSOICTSOICT0123456789012345678901234567890
123456789012345678901234567890123456789012345678
9SOICTSOICT

Bài tập 01.03 – Ứng dụng truyền nhận file UDP (v1)

- Ứng dụng **udp_file_sender** cho phép người dùng nhập tên file từ dòng lệnh, sau đó truyền tên và nội dung file sang **udp_file_receiver**. Địa chỉ IP và cổng của receiver cũng được nhập từ dòng lệnh.
- Ứng dụng **udp_file_receiver** nhận dữ liệu từ **udp_file_sender** và ghi vào file. Cổng chờ được nhập từ dòng lệnh.



Bài tập 01.04 – Ứng dụng truyền nhận file UDP (v2)

- Tại một thời điểm có thể có nhiều **udp_file_sender** cùng chạy. Để minh họa tình huống này, có thể dùng lệnh **usleep()** để tạm dừng chương trình trong một khoảng thời gian nhằm giảm tốc độ gửi file.
- Ứng dụng **udp_file_receiver** có thể nhận nội dung các file từ nhiều **udp_file_sender** khác nhau. Ứng dụng cần phân biệt nội dung file được gửi từ sender nào để thực hiện việc ghép nội dung file được chính xác.