

File Transfer Lab Exercises

ECE361

Student ID: _____

Date: _____

Due: 11:59 PM, October 6th, 2021

Instructions

This assignment is out of 20 marks, for a total of 2.5% of your final grade. It's not intended to be hard; it's intended to get you thinking about the labs (tip: you may find the File Transfer Tutorial PDF useful). Solutions will be posted to Quercus on October 7th. If (and only if) you receive a passing grade on this assignment, then you will have 1 week from the time your grade is posted to submit corrections, if desired. Corrections should also explain why your previous answer was incorrect.

Exercises

Socket Programming (10 marks)

1. Why does a server need to `bind`? Does it do this before or after it starts communicating with clients? **(2 marks)**
2. Does a client program also need to `bind`? If so, when does it do this? If not, why not? **(2 marks)**
3. A common misconception is that you need to call `recvfrom` before the sender transmits their message, otherwise you won't receive anything. Please explain why this is not the case. **(1 mark)**

4. Fill in the missing arguments to `recvfrom` below. (2 marks; 0.5 each)

```
char buf[1024];
struct sockaddr_storage their_addr;
socklen_t addr_size = sizeof(their_addr);
recvfrom(sockfd, A1, A2, 0, A3, A4);
```

A1 = _____

A2 = _____

A3 = _____

A4 = _____

5. The code below contains 3 issues or bad practises (that we know of). Identify them and explain. (3 marks)

```
int sockfd = socket(AF_INET, SOCK_DGRAM, 0);
struct sockaddr_in server;
server.sin_family = AF_INET;
server.sin_port = 54321;
server.sin_addr = "128.100.13.140";
socklen_t addr_size = sizeof(server);
char msg[] = {'h', 'e', 'l', 'l', 'o', '\0', 'h', 'i'};
sendto(sockfd, msg, strlen(msg), 0, (struct sockaddr*)&server, addr_size);
```

Serialization and Deserialization (10 marks)

Serialization refers to the process of converting a struct into a sequence of bytes in preparation for transmission. Deserialization is performed at the receiver and refers to the reverse process.

In general, it is not a good idea to pass structs directly into `sendto`. In the following exercises, we will ask you to think about why. Hint 1: why do we use functions such as `htons` and `htonl`? Hint 2: what issues might we run into when trying to send a pointer to another machine or process?

6. For each of the following structs, indicate whether it is okay to pass it directly into `sendto` and provide a brief justification. (2 marks each; 0.5 for correct YES/NO answer and 1.5 for justification)

<pre>struct Node1 { uint16_t byte; };</pre>	YES / NO	
<pre>struct Node2 { int age; char name[64]; };</pre>	YES / NO	
<pre>struct Node3 { int age; char* name; };</pre>	YES / NO	
<pre>struct Node4 { int data; struct Node4* next; };</pre>	YES / NO	

7. Is a Node3 harder to serialize than a Node4? Why or why not? (2 marks)