

Task Completion Status:

- Task 1 (Completed)
- Task 2 (Completed)
- Task 3 (Completed)

For task 1 I created a `isHostOnPort` function which sets the message content before sending the message until the host has received it and afterwards my device gives 50 attempts to receive their message, if unable to receive the message then the host is deemed not connected to that port. Then I tested the function by having two for loop going through all the host and through all 4 ports in the setup function, It worked perfectly however it did take a considerable amount of time for the message to be passed back and forth.

Task 1 instruction was clear and wanted to implement `isHostOnPort` function which will run the function and return a true or false statement which displays whether that host name is connected to port 2. When it came to testing I came across a complication where on the serial monitor, it kept displaying "Received RTS when sending RTS" occasionally when the program tries sending a message to a host through port 2, this error really confused me because it means that whenever my device is trying to send a message to the host, its also trying to send a message back but the main problem it caused was that it kept looping the same error since it kept blinking my Arduino's yellow LED and printing out the same error text on the serial monitor. At first I thought I could have my device to check whether any message are being sent its way before sending the message however this doesn't solve the issue because even though my device has checked any incoming message before sending out their message, it doesn't stop the host from sending a message the moment my device sends a message. After some tinkering with the code I set the variable "trials" designed for the amount of tries to fetchMessage from the host from 3 to back to 50 which was altered after I found out after testing the `isHostOnPort` function took a considerable amount of time and I didn't want to wait for my program to take so long to display the results which is why I changed the value from 50 to 3 to speed up the process.

After the change I ran task 1 function and it worked perfectly. Turns out from what I have gathered by speeding up the process from the amount of attempt to only 3 it gives the host only 3 windows open to send their message across to our device but sometimes our device didn't get the message and then starts sending a message to the next host however at some point the previous host has caught on and starts sending RTS messages to our device which collides with our device trying to send a message to the next host and carries on sending RTS until the device responds back.

For task 2 I created `getPortForHost` function which include a for loop starting from 1 to 4, every iteration of another loop it will grab the host name and it goes through `isHostOnPort` function with parameter of the host name and a value from the loop (ranges 1-4 which represents the ports). If the `isHostOnPort` returns true then it will return the loop value. However, if the loop finish then it will return 0. The return value represents the host name connected to x port. I tested this function by having a loop which goes through all the host names and putting it in the `getPortFoHost` function and print out the value. It works as intended.

Another complication that I have faced was initially I wanted to create two array, one of all 26 alphabets and another which contain 26 LEDs array. So I would have each character to be checked whether it mates with one of the elemt of the alphabet array however I deemed this as totally inefficient because worse case scenario it would go all the way to "Z" in the alphabet which is extremely time consuming. After some thinking I remembered in Worksheet 2 of networking practical it specified that char array consist of numbers which ASCII values. Through some testing "A" is the 65 value in char array therefore I can sort each letter of host name into 1-26 by subtracting 64.

As for task 2 function I created a loop which iterates through all host names. For each iteration it will be passed through getPortForHost function and then return the value of which will be put through an if statement which will check if the value of that variable is not 0, if condition is met then it will be passed through the blink_network_name_leds function.

Inside the function contains an array of 5 LED lights. It checks if the argument is divisible by 5 bit binary of which each one represent an LED light that will be turned on. For example H is the 8th letter of the alphabet and will light up the pink LED.

For task 3 it wanted me to create a function getTimeToHost, inside I had used a milis function to store the runtime and passes the arguments to the isHostOnPort function, if that function returns true then the current runtime will be grabbed and be subtracted by the runtime previously stored in to get host response time.

As for the task 3 function a loop that goes through host name is passed through getPortForHost, if a port is assigned to that host then another loop which goes through 50 times which utilizes getTimeToHost function which will return the response time of that host of which will be checked if its current the fastest and slowest of that host. After 50 iteration the fastest and slowest time will be displayed. Task 3 functions works as intended.

I believe if I were to award my work it would be within 70-79% range because of the efficiency and layout of the code.