Network Programming Assignment 2, Design Document: Q1

Student Name	**Student ID**
Anirudh Buvanesh	2016B4A70614P
Ashish Kumar	2016B4A70636P

The problem requires us to design a ping utility which when supplied with multiple IP addresses (IPV4/6) pings each IP three times, purpose of which is to measure the RTT.

1. Problem variables

a. **BATCHSIZE:** Defines the size of the window, i.e at any given instant there will not be more than **BATCHSIZE** number of ping requests that are being handled. This is done in order to keep the design scalable. Here we assume the value as 32.
b. **PINGS:** Defines the number of pings that need to be done for a particular host, here we assume the value as 32.
c. **SEC_TIMEOUT, MICROSEC_TIMEOUT**: Defines the maximum amount of time we are willing to wait for a ping response to arrive. If the response doesn't arrive in the specified time, then we print the ping responses obtained for that IP till now and do not perform subsequent pings.

2. Problem design

The problem maintains the structure (*pr*) which is an array of structures of size BATCHSIZE that stores relevant information for handling of a particular ip (*struct proto in rtt.h*). In the beginning a batch of IPs are fetched and the first ping is sent for all of them, when we receive a response for any of the pending IPs we check if we have received the last ping or not, if yes then we report the RTT for all the ping requests, if not we send the next ping request.

Two raw sockets are maintained for handling IPv4 and IPv6 addresses and a select call is used to monitor these 2 fds. Select is used with a timeout which is controlled by above mentioned parameters. In case the select returns 0 (*case where no activity on both sockets*) then we check if timeout has expired for any of the IPs, if yes then the procedure mentioned in 1 c. is executed.

3. Performance statistics

We tested on a set of 22 IP addresses and performed 3 pings on each IP address and obtained a throughput of *19.825 ips/sec*

Files submitted

rtt.c - source code (*implementation logic over here*)

rtt.h - header file that contains header files for the corresponding source code

makefile - generates the executable rtt.o
Instructions.txt - provides the instructions for executing program
ips.txt - Sample IP addresses for testing