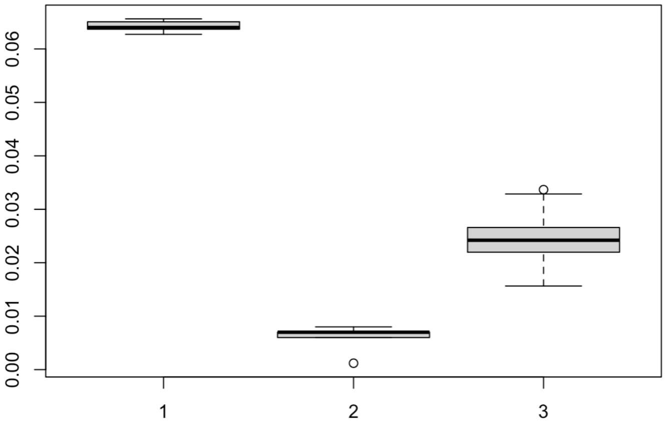
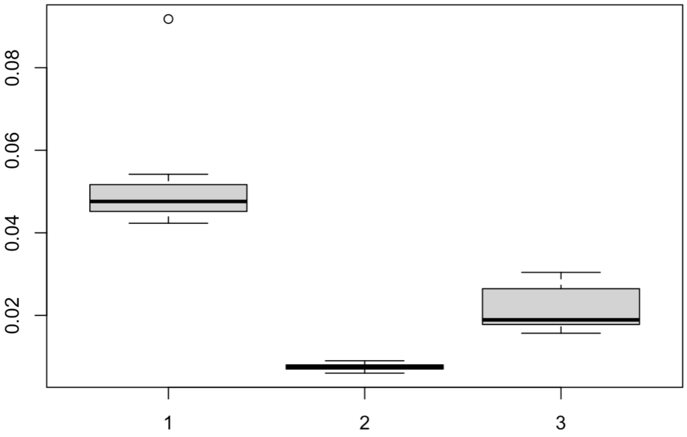
The libraries that I used for my program were the dns and datetime libraries. I used multiple modules from the dns library including name, rdataclass, rdatatype, message and query. All of these modules were necessary for me to resolve the queries for the domain names. I used the datetime library to find the total time it took for my dig to run. In order to run my program all you have to do is enter the domain name as the input. The output that we get is seen in the mydig_output file.

Part B:

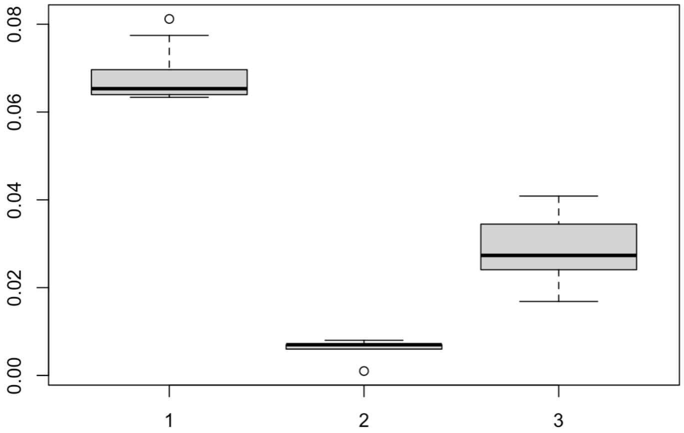
APPLE



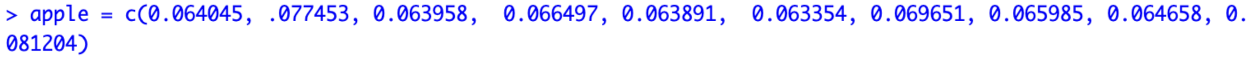
FACEBOOK



NETFLIX

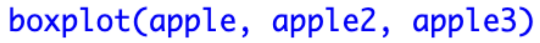


I was able to get the boxplots shown above using R. I first inputted all the data that I received from the trials shown below into R. This can be done using the command shown below.



```
> apple = c(0.064045, .077453, 0.063958,  0.066497, 0.063891,  0.063354, 0.069651, 0.065985, 0.064658, 0.
081204)
```

To get the boxplot for this data we would use the command shown below.



```
boxplot(apple, apple2, apple3)
```

We did this with all 3 websites

From the results shown in the box plot charts shown above, the my_dig program using the root server had the worst runtime (1), the local dig did the best (2), and the my_dig program using the google dns did the second best (3). The averages with the my_dig program using the root server were 0.0680696, 0.0522821, 0.0642463 for apple, facebook, and netflix respectively. The averages with the local dig function were 0.0062, 0.0074, 0.00642 for apple, facebook, and netflix respectively. The averages with the my_dig program using the google dns were 0.0277611, 0.0212884, 0.024365 for apple, facebook, and netflix respectively. Based on the boxplots of all 3 experiments the experiments run with the my_dig program had a wider range of results whereas the experiment run with localhost had a much smaller and consistent range of runtimes. The experiments run with the my_dig program also had slower runtimes in general. There could have been multiple reasons as to why the my_dig function was so much slower in comparison. The first reason could be due to network latency. Since we are performing a dns query over the network, if the network is slow then the program will also have a slower overall runtime. The second possible reason could be due to server traffic. The program runtime also depends on how bad the traffic of the dns server is. Finally the last reason is based on how many times the program is recursively calling itself until it reaches the bottom layer for an answer. As seen from experiment 1 an 3, we saw that apple had a longer runtime from all 3 experiments compared to netflix. This means that apple had more layers to dig through compared to netflix. Overall, if I were needed to use a dns resolver, I would rely on the local dig function through the terminal.

Experiment 1: my_dig
APPLE
1: 0.064045 seconds
2. 0.077453 seconds
3. 0.063958 seconds
4. 0.066497 seconds
5. 0.063891 seconds
6. 0.063354 seconds
7. 0.069651 seconds
8. 0.065985 seconds
9. 0.064658 seconds
10. 0.081204 seconds
FACEBOOK
1. 0.044873 seconds
2. 0.048041 seconds
3. 0.047141 seconds
4. 0.046345 seconds
5. 0.045179 seconds
6. 0.05168 seconds
7. 0.091793 seconds
8. 0.05418 seconds
9. 0.04232 seconds
10. 0.051269 seconds
NETFLIX
1. 0.065627 seconds
2. 0.065553 seconds
3. 0.065082 seconds
4. 0.062734 seconds
5. 0.063688 seconds
6. 0.063867 seconds
7. 0.063635 seconds
8. 0.063902 seconds
9. 0.064258 seconds
10. 0.064117 seconds

Experiment 2: local dns

APPLE

1. 6 msec
2. 10 msec
3. 6 msec
4. 7 msec
5. 8 msec
6. 7 msec
7. 7 msec
8. 7 msec
9. 6 msec
10. 7 msec

FACEBOOK

1. 9 msec
2. 8 msec
3. 8 msec
4. 6 msec
5. 7 msec
6. 7 msec
7. 8 msec
8. 6 msec
9. 7 msec
10. 8 msec

NETFLIX

1. 7 msec
2. 12 msec
3. 7 msec
4. 8 msec
5. 7 msec
6. 7 msec
7. 6 msec
8. 8 msec
9. 7 msec
10. 6 msec

Experiment 3: Google public dns
APPLE
1. 0.028503 seconds
2. 0.034471 seconds
3. 0.028647 seconds
4. 0.035099 seconds
5. 0.018577 seconds
6. 0.024079 seconds
7. 0.040861 seconds
8. 0.016845 seconds
9. 0.02618 seconds
10. 0.024349 seconds
FACEBOOK
1. 0.016671 seconds
2. 0.019293 seconds
3. 0.018437 seconds
4. 0.018536 seconds
5. 0.028538 seconds
6. 0.030413 seconds
7. 0.017796 seconds
8. 0.015673 seconds
9. 0.026461 seconds
10. 0.021066 seconds
NETFLIX
1. 0.022393 seconds
2. 0.026596 seconds
3. 0.024575 seconds
4. 0.016986 seconds
5. 0.023842 seconds
6. 0.033675 seconds
7. 0.015647 seconds
8. 0.021966 seconds
9. 0.025103 seconds
10. 0.032867 seconds