Module 3: Mapping with *traceroute*

Module: Quick Look

Abstract:

Networking Infrastructure is an important concept that many high school curriculums do not cover. One of the easiest ways to present students with Networking Infrastructure concepts is to discuss ways of how data is communicated over the Internet. The purpose of this module is to provide structure for a teacher to be able to allow students to program their own map using a commonly found function on computers, *traceroute*; following the path that a packet takes from sender to receiver.

Intended Audience:

- Students who have completed first semester of AP Computer Science (or equivalent)

Material Requirements:

- Programming language of instructor's choice installed on computer (suggestions for IDE in the "Module: Suggestions" section)

Total Instructor-Preparation Required:

- 30 minutes of working with *traceroute* diagnostic tool and program files included in this module
  - opening & analyzing the purpose individually and the interconnectedness of all as a process
- 60 minutes of writing own program
  - based on your comfort level, you may want to write program using your language of choice if different from what is shared with you
- 40 minutes planning

Total Class-Time Required:

- 120-200 minutes (includes instructional time and programming time for students)

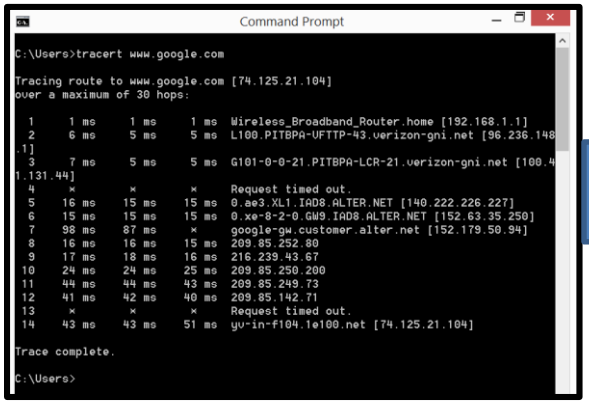
Key Terms:

- **Sender**
- **Receiver**
- **Domain Name**
- **DNS Lookup**
- ***traceroute***
- **IP address**
- **IP number**
- **Regular expressions (pattern recognition)***
- **Latitude/Longitude**
- **Mercator Projection Map**

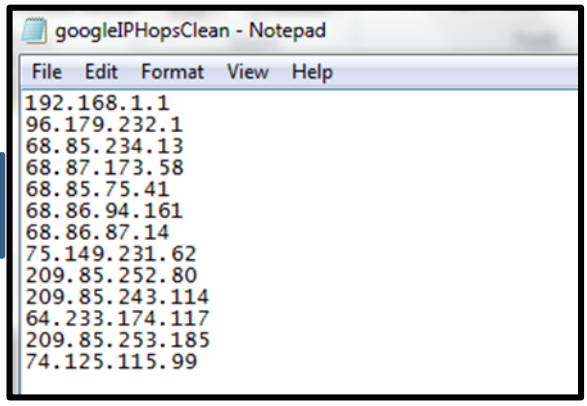
** if this is new concept for students*

Process Overview

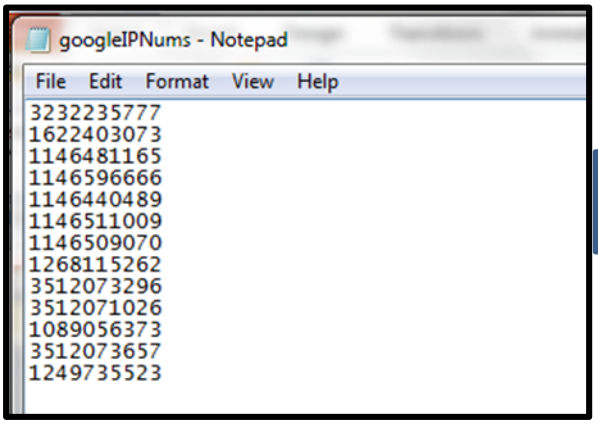
The following diagram illustrates the artifact of each step of the process. A further description of each step is found later in this module.



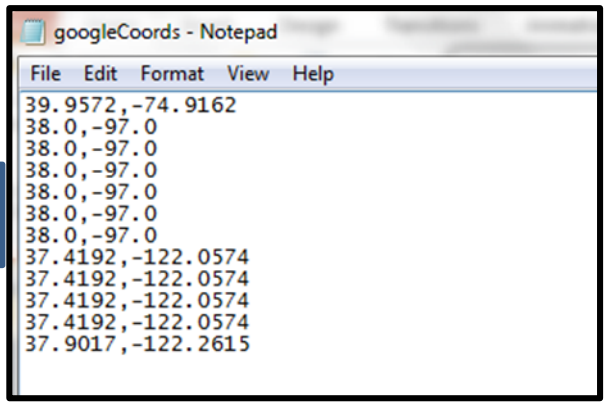
Step 2
Results of *traceroute* tool



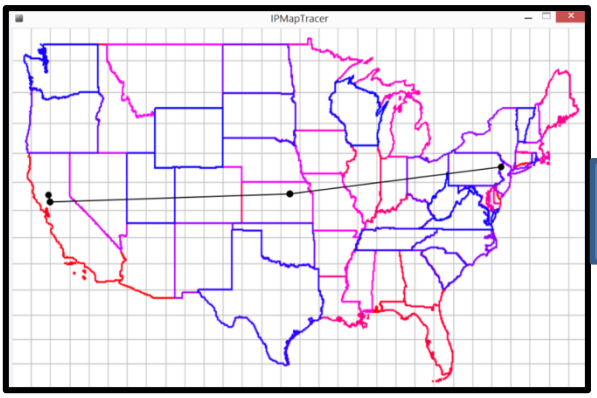
Step 3
Created text file of IP addresses



Step 4
Created text file of IP Numbers



Step 5
Created text file of Latitude/Longitudes



Step 6
Graphical Illustration of Path

The following resources are available in the module to assist you with preparing to present the topic in class:

- **Module: Suggestions** prepares you to use the module and developing your lesson plan.

- **Module: Scope & Sequence** provides you with the objectives for your students, the basic instructional content, and ways to have students demonstrate their understanding of the protocol through a project.

- **Module: Extensions** identifies ways for you to enrich or modify the suggestion to better meet the needs of your students.

Module: Suggestions

The following are suggestions to assist you in preparing for use of this module:

- After reading through the entire module, plan your unit of study based upon your comfort with module content, your ability to program and/or teach programming, and how much available time you have in your schedule.

- If necessary, utilize the supporting materials accompanying this module (Module3_Resources.pdf) to increase your understanding of the *traceroute*. You could also provide students with this information as needed.

- Determine IDE (development environment) and programming language to be used:
    - Processing
        - https://processing.org/
    - Eclipse: JAVA
        - https://eclipse.org/downloads/packages/eclipse-ide-java-developers/keplersr1
    - Other: This module provides only suggestions so feel free to use what you like:
        - Python, Scratch, C, etc.

    For this assignment, if you choose a different development environment then what you currently teach, you may need to factor in some additional instructional time.

    Also, depending on your focus of graphics throughout the Computer Science course, you may need to review graphical applications including but not limited to JFrame, Graphics, and Graphics2D classes. In addition, you will want to make sure that if your students have not learned how to open, read, and save to a text file that you provide them with proper resources to do so. In this module, when programming with Java, the PrintWriter class was used.

- If you like the concept(s) of this module but do not have sufficient amount of time to complete every step, you can:
    - work through as little as you like by sharing files with your students
    - provide the students with access to the files and have them describe the purpose of each file and how they work together
    - give the students a text file that contains latitudes/longitudes and have them write just a program that graphs the route of the packet ~ see step 6
    - write all of the programs in one file; we chose to write them in separate files to show specific purpose of each step within the process

Module: Scope & Sequence

Suggested Timeline:

Step 1: 10 – 15 minutes ~ Present & discuss *traceroute*, domain names, IP address & IP numbers

Step 2: 10 – 15 minutes ~ How to use the *traceroute* tool

Step 3: 30 – 40 minutes ~ Code: IP Address

Step 4: 15 – 25 minutes ~ Code: Convert IP addresses to IP numbers

Step 5: 20 – 60 minutes ~ Code: Convert IP numbers to Latitude/Longitude

Step 6: 40 – 80 minutes ~ Code: Graphical Illustration of the *traceroute* Path

---

Teacher should present the following information to students:

Learning Objectives:

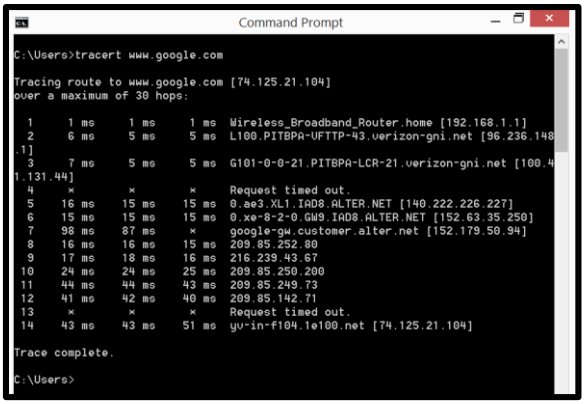
Upon completion of this module, the student will be able to:

- Define *traceroute*
- Identify the components involved and explain their role
    - sender, receiver, packet, domain name, DNS Lookup, IP address, IP number
- Produce a program that creates a text file of a list of IP addresses associated with a route to a specific website
- Produce a program that creates a text file of converted IP address to IP numbers
- Produce a program that creates a text file of latitudes/longitudes that corresponds to IP numbers
- Produce a program that graphically draws a visualization of a path on a picture of a map of the United States that a packet travels from sender to receiver

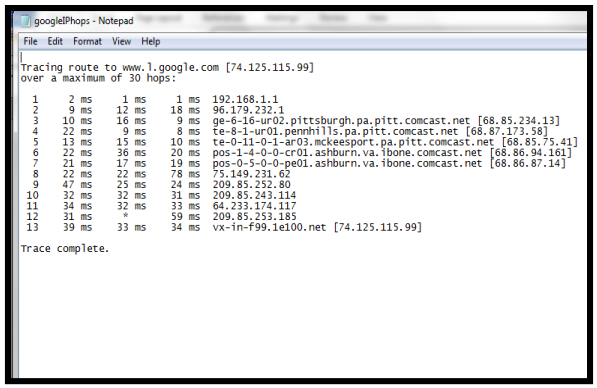
Sequence of Instructional Content:

1. **Introduce the concept of tracing a packet's route**
    - Define for students the following: IP address and DNS
    - Discuss the purpose of a domain name and how it relates to an IP address. Also, this is a good time to discuss DNS Lookup. Additional information is available in the Resource Section of this module.
    - There are plenty of websites that can be used to demonstrate visually depicting *traceroute* or you can leave it a secret and let the students be surprised of what they have accomplished when they are done with the final program.
        - **i.** Monitis – shows a path to a website of choice but from their server's location not your computer's IP address

2. **Determine IP addresses of switches/routers/servers a packet follows from sender to receiver**
    - Discuss what is necessary to track the path of a packet, e.g. if you wanted to track the path of a packet along switches/routers/servers from a computer to www.google.com this would require ~ *IP addresses, physical locations of web servers/routers, and a way to convert IP addresses to IP numbers to latitude/longitude*



    - *traceroute tool note: Have students conduct the following steps on a school computer or personal computer* to guarantee that they receive the correct path data.*
      based on your school's wireless network structure, you may need students who have brought their own device to "hard-wire" connect** to the network if possible using an Ethernet cable.*
    - **Utilize the *traceroute* tool**:
        i. ***Windows*** – The following is the basic command to use the *traceroute* tool in the command line of a Windows machine: **tracert www.url.com**

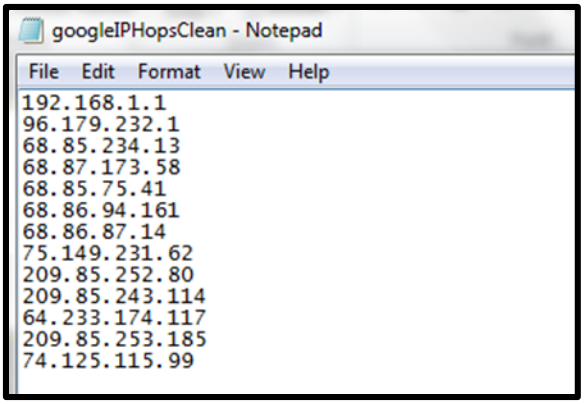


            1. For additional information to assist you with this process, click here.
            2. To create a text file of the *traceroute*, use the following command:

> **tracert www.url.com > C:\Users\results.txt**
> tracert website > location of where to save the file\filename.txt

        ii. ***Mac-*** The easiest way to use the *traceroute* tool on a Mac is to use "Network Utility," click "Traceroute," type in the url, and then click on "Trace." Once the results appear, to save them as a text file, they will need to be copied and pasted into a text editor and then saved as a .txt before moving onto the Step 3.
            1. For additional information to assist you with process, click here.

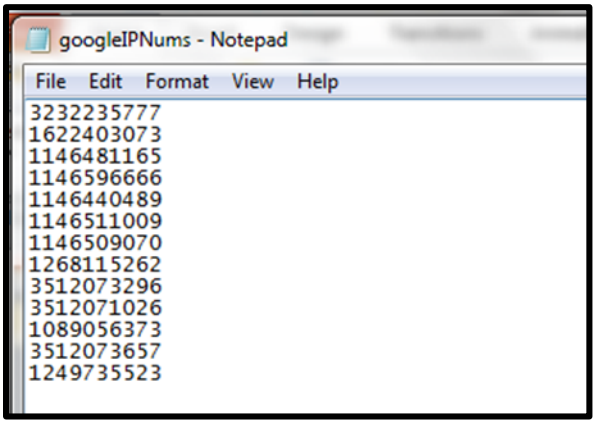
3. **Write a program that searches a text file with IP addresses and other information and saves them to another text file that only contains IP addresses**
    - At this point, the students need to work through writing a program that searches a text file of the output for *traceroute* for the IP address that was saved/copied from the



previous step. In the end, only a file that contains all of the IP addresses should be created (we must eliminate extra data that will not be necessary to map the locations). During this process introducing the concept of Regular Expressions (Pattern Recognition), if you haven't done so already, could be helpful for your students to search for IP addresses. Take a look at the attached file to see the pattern used. Also, be careful to remove the first instance of an IP address as it points to the final destination and is not needed.

- One version of this program is seen as "Java program that strips the IP addresses from the traceroute command's output [java]" under "Additional Resources" on webpage.

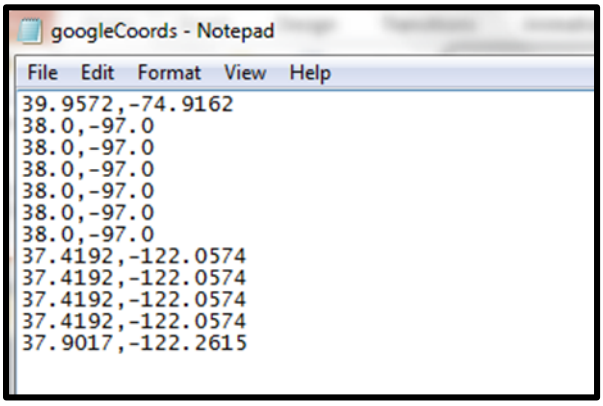
4. **Write a program that converts IP addresses to IP numbers and saves them to a text file**
   - To prepare to search a database that correlates an IP address to latitude/longitude, the students will need to write a basic program that does the following:
     i. Opens and reads in text file from previous step
     ii. Converts an IP address to IP number using the following formula:



ipnum = 16777216*w + 65536*x + 256*y + z
where IP Address = w.x.y.z

     iii. Creates a text file with IP numbers that corresponds to the list of IP addresses from previous step
   - One version of this program is seen as "Java program that converts IP addresses to IPNumbers for querying database [java]" under "Additional Resources" on webpage.

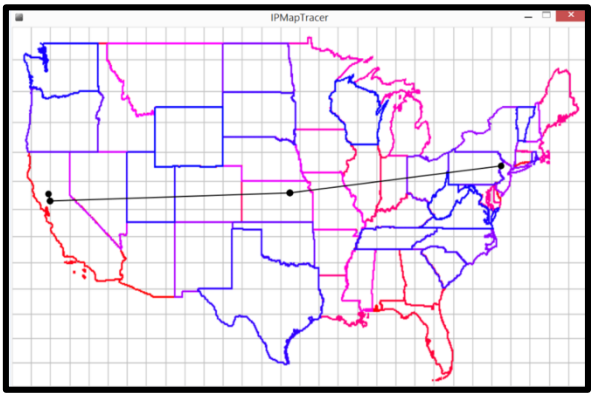
5. **Write a program that creates a list of latitude/longitude coordinates corresponding to the IP numbers from previous step and saves them to a text file**
   - Now that the IP numbers are available, a program can be written to search through a database to find the corresponding latitude/longitude coordinates.
   - This is by far the most difficult program to write within this process. You may decide to just give it to your students and have them determine what is happening and update it to match their filenames ~ this approach has been used by many teachers in the past.



For students who like a challenge, you may want to have them write this program on their own ~ this approach has also been used by teachers in the past with success. Either way, you will need to share the two databases ("First part of database for looking up IP Numbers [csv]" and "Second part of database for looking up Latitude/Longitude coord. [csv]") under "Additional Resources" on webpage.

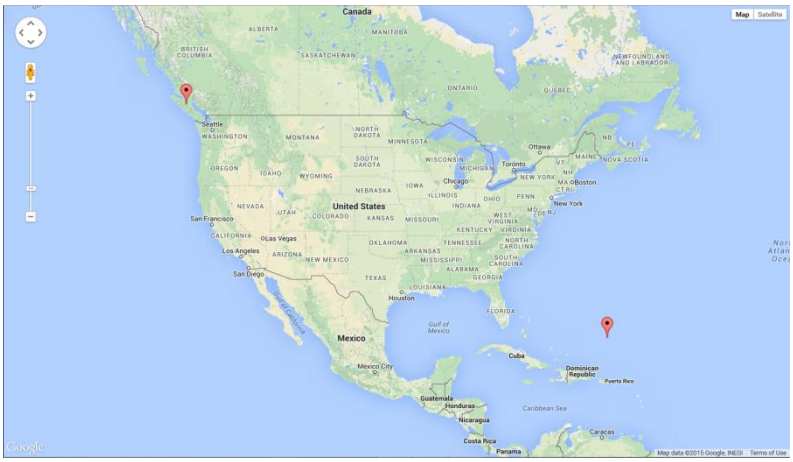
- *Note: the databases are a few years old but work well. You may find an IP number that doesn't correspond to a location and this is fine. You can search for a new database but will need to make sure that the attached file is rewritten so that it matches what you are searching for in the new file. This could be a timely venture if you decide to change databases.*
- One version of this program is seen as "Java program that looks up IP Numbers in database and returns list of latitude and longitudes [java]" under "Additional Resources" on webpage.

6. **Write a program that graphically illustrates the path a packet follows using the latitude/longitude coordinates of IP addresses**



   - Now that the students have a file that contains the latitudes/longitudes coordinates for a packet's full route traversal from computer to website server, they can use that data to place dots on a United States map at the proper locations.
   - To receive the most accurate results, students should use a Mercator map
   - When programming this part of the project, using or creating a map()* function (depending on the language being used) could be helpful. In the example provided, we used Processing to draw our path. To assist, we used Google Maps to help us find the latitude/longitude of the northwestern most part of the United States and associate the latitude/longitude coordinates with the pixel coordinates of the chosen United States map image.

     **this map() function is available in Processing or can be written by the student and doesn't correspond to any instance of the word "map" used in this module to describe geography*



- *Note: when working with the latitude/longitude, students should take into consideration the order in which latitude/longitude is used within their program. Sometimes the order is reversed and will provide an incorrect illustration of the path (i.e. latitude comes first when expressing a coordinate and longitude comes second; latitude represents the up/down on the map or y-coordinate, and longitude represents the left/right on the map or x-coordinate**)*
  ***in Mathematics, the x-coordinate is listed first in ordered pairs which can be a source of confusion for students if not noted*
- One version of this program is seen as "A Processing program file that renders a map from Latitude/Longitude coord [pde]" under "Additional Resources" on webpage.

Guiding Questions/Statements

- What is a domain name and how does it relate to an IP address?
- Describe what would be necessary to track a packet sent from a sender to receiver.
- Differentiate between IP address and IP numbers.
- Why is a Mercator map used?

Mapping with *traceroute* Project:

For this module, there is no specific summative assignment to reinforce this multiple step process that was presented and completed by the students. The intentions of this module were to allows students to work through the process of writing multiple programs that work together to prepare them to write a final program that graphically illustrates, with dots and lines on a map, the path a packet takes from sender to receiver. Those teachers looking for an assessment have often taken the last step and turned that into a student-driven project requiring students to simply create the dots on the map that correspond to IP addresses that a packet follows from computer to website server.

The following assessment tool can be used to evaluate the students coding project:

This can be found under the filename: TracerouteRubric.xlsx

Module: Extensions

- Students can rewrite the final program in Step 6 to take into account a web address that follows the path of a packet outside of the United States (e.g. www.wikipedia.com at times routes through Europe). Within Module 2 content, a world Mercator map has been included.