CST8130: Data Structures --- Assign #5- Routing Table

Using ArrayList(Assign2) or Dyamically Allocated Array(Assign1) with Hashing Data Structure

DUE: Wednesday December 9th by 10PM SHARP!

Problem Description:

In this assignment, we will update either Assignment #1 solution (working with dynamically allocated array) or Assignment #2 solution (working with ArrayList) to implement a Hashing algorithm for storage of destination address into the routing table.

Note – you will use as your hash algorithm the value of the third octet of the destination IP address modulus the maximum size of the array or ArrayList.

We will not have a collision resolution – if the space is already used, you do not add the destination network address to the array/ArrayList. (Note – this simplified approach is for time reasons – the best thing for us to have done is to have a linked list that RoutingTableEntries get inserted into at each cell – you are welcome to take this approach for extra marks).

This means that you will have to initialize all your array/ArrayList cells to null, and you will have to check for the cell == null before you try to use the data in it (for printing the routing table for example). You must change the displaying of routing table entry to show the index that the entry is stored at.

Sample Run:

```
Enter number of entries maximum for
array: 50


Enter name of file to process:
c:\trafficpackets.txt

Adding entry to routing
table192.168.1.0\24
at index 1
Sending packet out e0
192.168.1.0\24
Adding entry to routing
table192.168.4.0\24
at index 4
Entry is already in the routing
table192.168.4.0\24
Sending packet out e1
192.168.4.0\24
Adding entry to routing
table192.168.3.0\24
at index 3
Sending packet out s0
192.168.3.0\24
dropping packet....192.168.0.0\16

Routing table...

[1] 192.168.1.0\24 Port: e0
[3] 192.168.3.0\24 Port: s0
[4] 192.168.4.0\24 Port: e1
```

Submission:

You must submit to the assignment link in Blackboard by the due date and time a zip file (named LastnameFirstNameAssign5) containing:

- all source code – ie .java files (Note – I may choose to re-compile your program….so all code must be available to me) with
- You can basically test your program with the same test plan used for Assignment 1 or Assignment 2…submit a test plan for any additional test cases you need to run.

Failure to provide any of the above will have an effect on your grade for this assignment. Marking guide will be published shortly.