Design an application router that has two functions: loadConfig() and findRoute(input) -> server. You can use any programming language of your choice.

**loadConfig():**

Load config is called only once during the application start up. This function parses and loads the routing config file (example given below) into an internal data structure used by findRoute() function. Once loaded, the routes do not change. Routes are specified as one pattern per line (see the example below).

**findRoute(input) -> server:**

findRoute takes a string as an input and returns a server name that is also a string. Input is a string like "<customer\_id>.<country>.<state>.<city>". findRoute function returns the server that matches the input. The routes in config file could contain ‘\*’ in them to accept any value. ‘\*’ appears only in the trailing parts of the rule. Rules are not given in any specific order in the configuration file. When more than one rule matches, most specific match wins.

Sample routing rules from configuration file:

customer1.us.ca.\*=server1

customer2.us.\*.\*=server3

customer2.\*.\*.\*=server4

\*.\*.\*.\*=server5

customer1.us.ca.sjc=server2

The following are the sample inputs and return values against the config file specified above:

findRoute("customer1.us.ca.sfo") -> server1

findRoute("customer1.us.ca.sjc") -> server2

findRoute("customer2.us.tx.dfw") -> server3

findRoute("customer2.cn.tw.tai") -> server4

findRoute("customer10.us.ny.nyc") -> server5