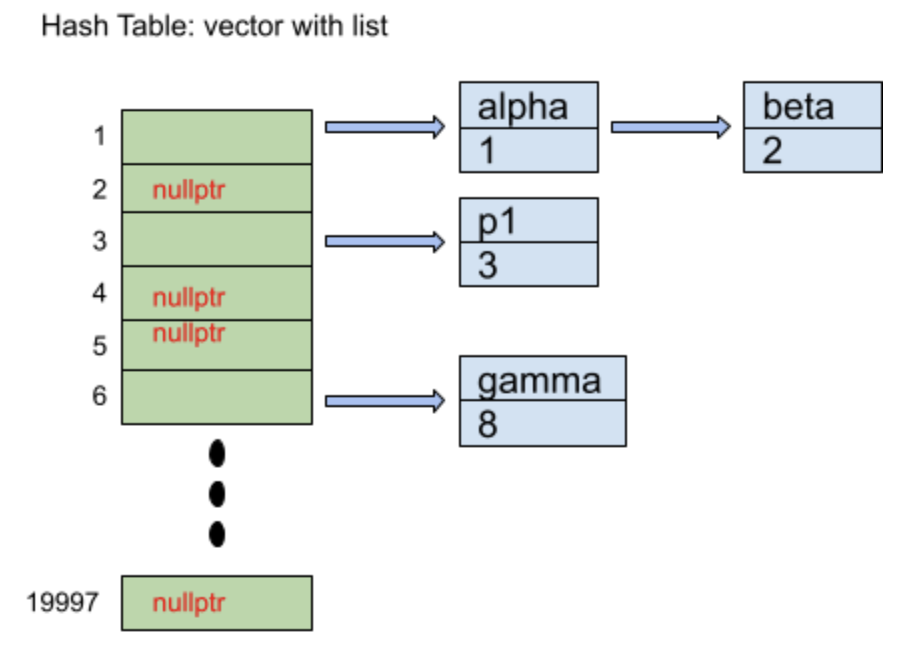
Aanas Chowdhury

UID: 905723833

Project 4

1. 

I chose to create a hash table, made up of a vector (The number of buckets was 19997) of lists (using the standard library implementation) because of how quickly it functioned to look things up in the vector and insert and delete items in the lists. The list is made up of nodes, which contain an id and a line number. Each function was accessed using a map function. I had another vector that also maintained the current scope and separated each scope with an empty string.

1. EnterScope: O(1)

ExitScope: O(N), N is the number of id’s in the list

Declare: O(N) N is the number of items in the list, since I am checking for previous declarations, adding something new however is O(1)

Find: O(N) N is the number of items in the list, but searching for the bucket in my has table is O(1)

1. Pseudocode

exitScope()

Repeatedly until we reach the end of the current scope

Get the bucket of the current id

Remove the most recently added id from the list in the bucket

Remove the id from the id vector

If the id vector is empty

Return false

Remove the scope entry marker

Declare(const string& id, int lineNum)

If the id is empty return false

Repeatedly until we go through the whole vector of ids

If we come across a scope entry marker

Break from this loop

If we find the id already in this scope

Return false

Add the id to our current scope

Create a new node in one of the buckets in the hashtable containing the id and the line number

Find(const string& id) const

If the id is empty return false

Get the bucket of the current id

Traverse that bucket’s list

If a node with that id is found

Return that node’s line number

Return -1 if nothing was found

1. It was very hard trying to figure out the scope. After multiple member variables and structures, I decided to just improve on what was given to us at the beginning. The slow implementation was actually surprisingly easy to go off of and using the empty string as entry markers were really helpful. Th was the main issue I had with this project. The implementation of the hash table was very doable.