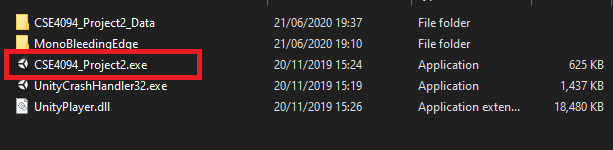
**Project Specifications**

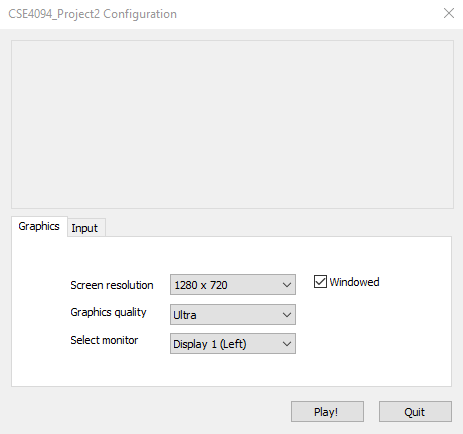
Cuckoo Hashing is implemented in C# language.

Interface is done with Unity Game Engine version 2019.2.14.f1.

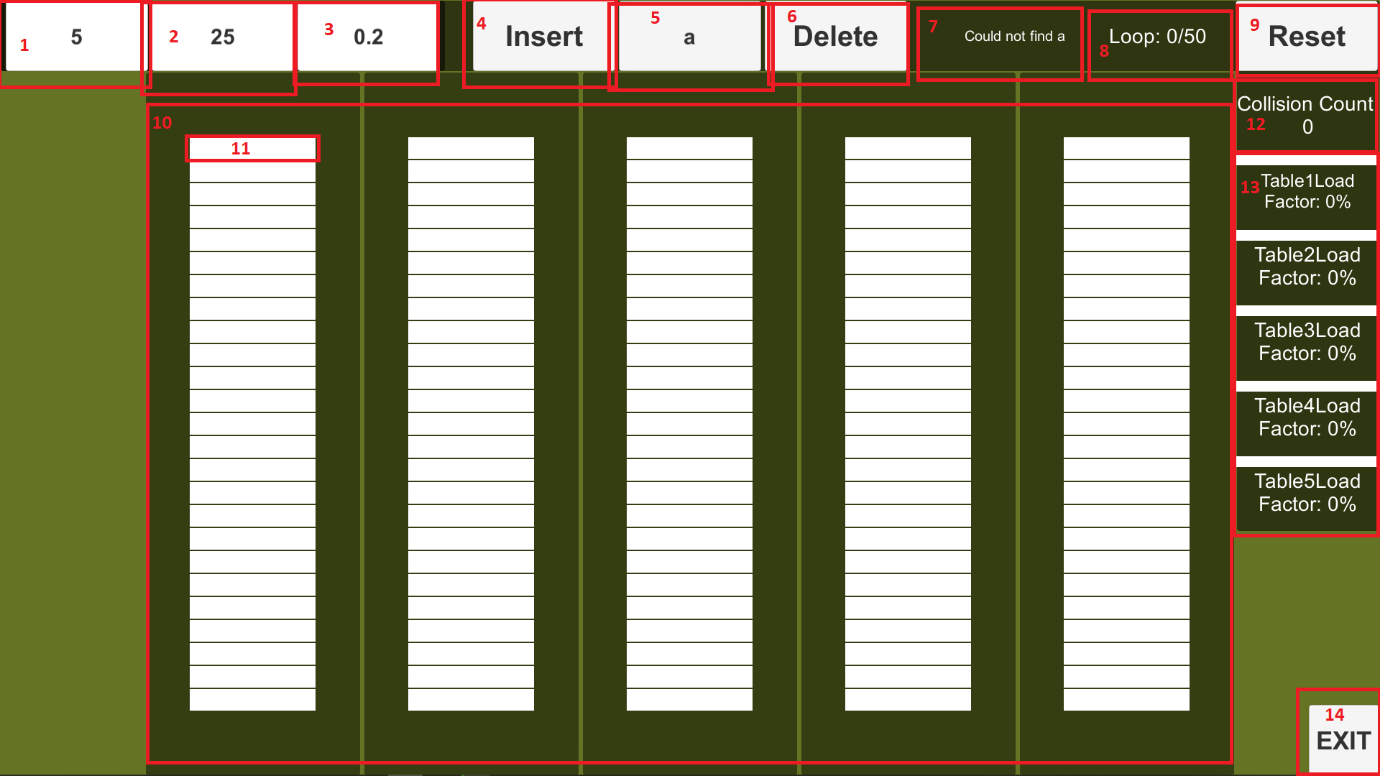
Run CSE4094\_Project2.exe.



Supported resolution is 16:9. I suggest you to run it in 1600:900. If you uncheck the “Windowed”, you can also set the resolution to 1920:1080 and run it full screen.



The interface will have the following parts:



**Interface Parts**

1. Enter the table count you want here. (integer only)
2. Enter the cell count per table you want here. (integer only)
3. Enter the waiting time between insertions here. (float, use “.” for decimal numbers.)
4. Inserts the word inside **5** according to Cuckoo Hashing result.
5. Enter the word you want to make executions on here. It will automatically search it. (type: anything)
6. Deletes the word inside **5**.
7. Info Text: Gives the info about the last instruction.
8. Loop Panel: Gives the info about the loop status.
9. Resets and clears all the cells.
10. Table Panel: Scales automatically with the input from **1**(1-5)
11. Cell Panel: Scales automatically with the input from **2**(10-30)
12. Shows the total collision count. Adds until resetting.
13. Shows the load factors of tables in % format. Scales according to the table count.
14. Use it to exit the program.

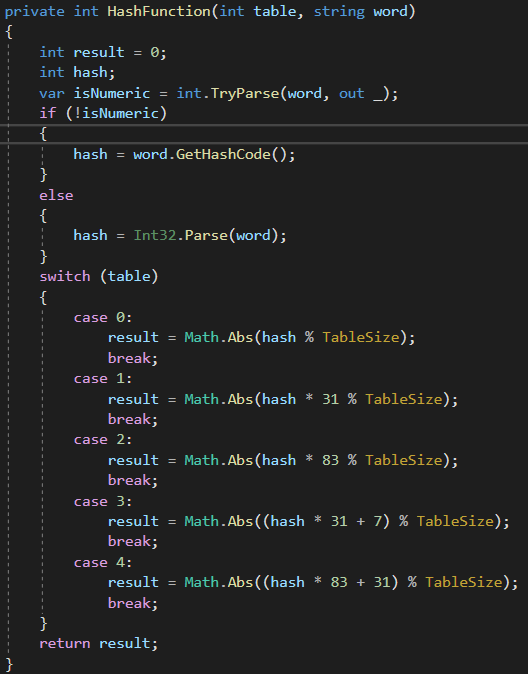
**About Implementation**

There are 2 main scripts(classes):

UIScript which handles the UI stuff.

CuckooHashing which handles the hashing.

EventManager will provide the communication between the scripts.

I’ve decided to use the [String.GetHashCode()](https://docs.microsoft.com/en-us/dotnet/api/system.string.gethashcode?view=netcore-3.1) because it’d be more efficient than anything I could’ve come up with. After generating an integer, I also feed it to a responding hash function of a table. It could be seen in the figure; 5 cases(tables), 5 different hash functions.

**Total number of collisions**

Collisions are calculated depending on the total loop count.

**Load Factor**

Load factor is calculated as (filled cells) / (total cells) for a table. And then converted to x% format.

You can also access the code from [here](https://github.com/alperenbayraktar/CSE4094_Project2).