COMP202 SPRING 2016

Programming Project

Due Date:23 March 2016 23:59

Hash Table Implementation

You are to write your own hash table in which both the keys and the values are of type **string** from scratch**. (Key as a string can contain letter (uppercase or lowercase), number and character).** The hash table is to use list-based map chaining to resolve collisions (Note that you are not allowed to use any java built in function/class of map).

The key and value pairs are stored in ***"input.txt"*** file. Read from the file and store your hash values into file named as ***"output.txt"*** file. Output file should include your hash table.
The linked list used by the chaining portion of the hash table must be written by you.

Your HashTable class should define following methods;

Constructors	default size of the hash table should be a prime (53)
hash (key)	Use the code given below. It should be your hash function. (it takes string key and calculate the hash)
size ()	Returns the number of key-value pairs in this hash table.
put(key,value)	Inserts the specified key-value pair into the hash table
remove (key)	Removes the specified key and its associated value from table
containsKey(key)	Returns true if a key is in the table otherwise false
keySet ()	return an iterable collection of the keys in the table
Empty ()	Clear everything in the hash table

Hash function :

Your hash function code is below.

```
int hash = 0;
int p=23;
                for (int i = 0; i < key.length(); i++)
                 hash = (p * hash + key .charAt(i)) % table_size;
```

Note : input.txt file contains key and value pairs in the each line. Each line till first space is key and the string after the first space is value (StringKey1 is key The End is value) and when the key and value pairs finish after first empty line some operation queries start. When you read from the file calculate hash values of the keys and put them into the **output.txt.** When you read a function perform the operation and write the result into output.txt. OutputtxtSample.txt is the output that you should provide.

If there is a collision, inserted key value pair should go to end of the list.(See Figure2)

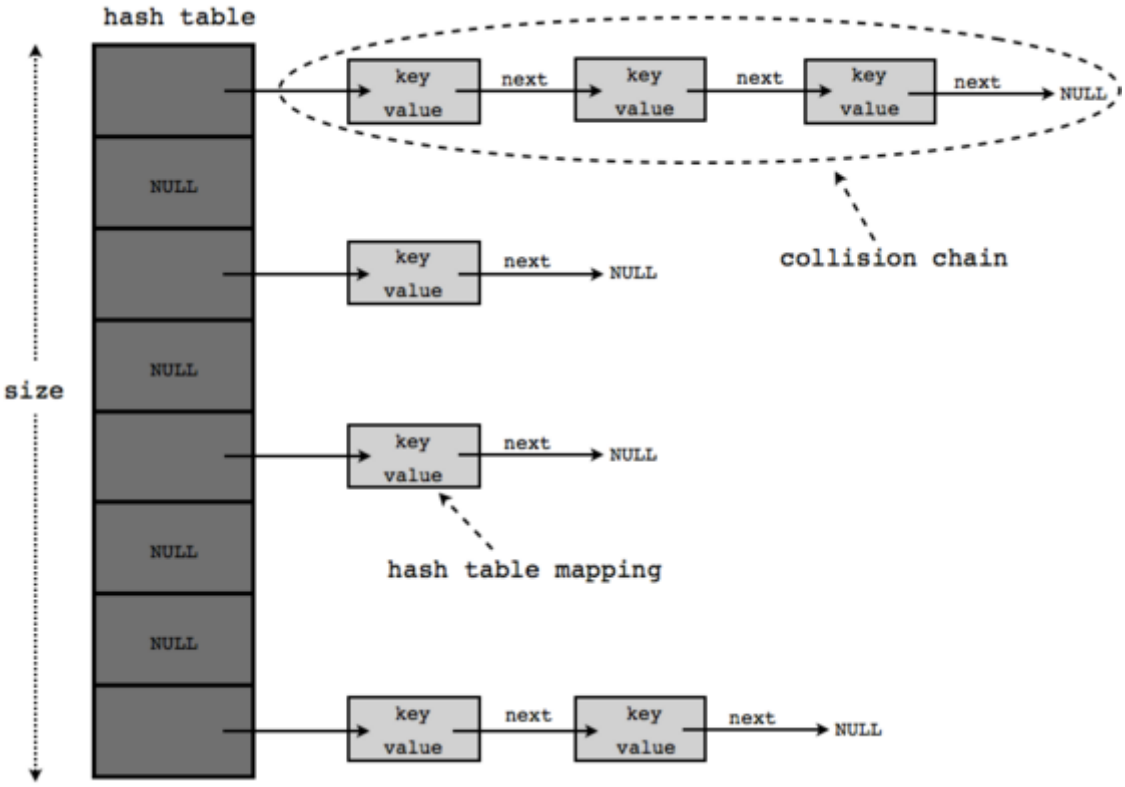

Figure1. Hash table with linked list

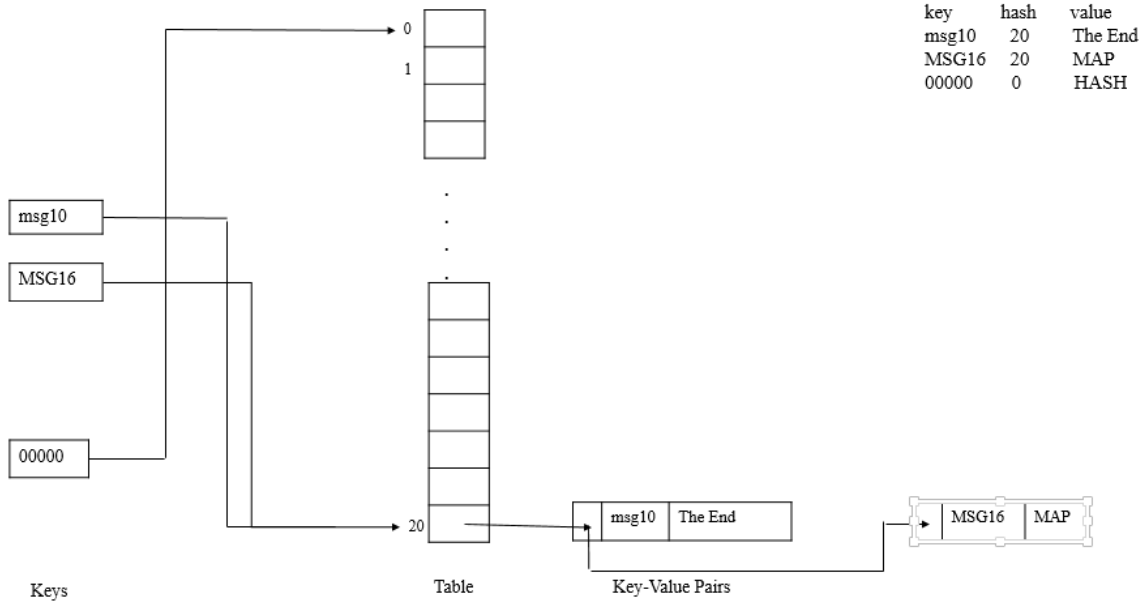

Figure2 illustrates put function

Rules:

1. This is an individual project. Any code help will be counted as plagiarism.
2. All the methods should be implemented from scratch. Which means any java library cannot be used.

Submission:

1. Does the program compile? (Programs that don't compile during the lab will not be graded)
2. Does the program meet all required interfaces?
3. Are comments well organized and concise?
4. If you have reviewed your code and all the criteria on the checklist are acceptable, follow the submission procedure.

!!!!!! Important

Please **do not zip your files** and follow the submission rules the projects uploaded different than given format **will not be graded.**

Upload it through Novel Login to the

F drive/COURSES/UGRADS/COMP202/PROJECT/YOUR_NAME

Please consider the following format for your projects submission:

Project2 "This should be only the name for your folder"

Do not upload two versions of your project under the same name, instead keep a counter and add it to the end of your file:

first version : Project2
second version: Project2_V2
ith version: Project2_Vi

We only check the latest version of your submissions except for exceptional cases. Please try to upload JUST one and the final version.