

323.25 Final Q1

Student: Jingshi Liu

Due Date: 5/19/2022

Source Code:

```
//  
//  main.cpp  
//  323Final_Q1  
//  
//  Created by Jingshi Liu on 5/19/22.  
//  
  
#include <iostream>  
#include <fstream>  
using namespace std;  
  
class ListNode{  
public:  
    string data;  
    ListNode* next;  
  
    ListNode(string data, ListNode*next){  
        this->data = data;  
        this-> next = next;  
    }  
  
    string printNode(){  
        string res = "(";  
        res += data + ", ";  
  
        if(next){  
            res += next->data + ") -> ";  
        }else  
            res += "NULL) -> ";  
  
        return res;  
    }  
};  
  
class HashTable{  
public:  
    string data;  
    int bucketsize;  
    ListNode** hashtable;
```

```

HashTable(int bucketSize){
    this->data = "";
    this->bucketSize = bucketSize;
    hashtable = new ListNode*[this->bucketSize];
    for(int i = 0; i < this->bucketSize; i++){
        hashtable[i] = new ListNode("dummy", NULL);
    }

    int hashString(string data){
        unsigned int val = 1;

        for(int i = 0; i < data.length(); i++){
            val = val*32 + (int)data[i];
        }
        return val % bucketSize;
    }

    void hashInsert(int index, string data, ofstream& outFile1,
ofstream& outFile2){
        outFile2 << "***enter hashInsert method ***" <<endl;
        ListNode* spot = findSpot(index, data);

        if(spot->next && spot->next->data == data)
            outFile1 << "*** Data is already in the hashtable, no
insertion takes place" <<endl;
        else{
            ListNode* newNode = new ListNode(data, NULL);
            newNode->next = spot->next;
            spot->next = newNode;
        }
    }

    ListNode* findSpot(int index, string data){
        ListNode* spot = hashtable[index];

        while(spot->next && spot->next->data < data){
            spot = spot->next;
        }

        return spot;
    }

    void printList(int index, ofstream& outFile){
        outFile<<"HashTable["<<index<<"]: ";
        ListNode *pointer = hashtable[index];
        string printing = "";
        while(pointer){
            printing += pointer->printNode();
            pointer = pointer->next;
        }
    }
}

```

```

    }
    outFile << printing << endl;
}

void printHashTable(ofstream& outFile){
    for(int i = 0; i < bucketsize; i++){
        printList(i, outFile);
    }
    outFile<<endl;
}

};

int main(int argc, const char * argv[]) {
    ifstream inFile;
    ofstream outFile, debuggingFile;
    inFile.open(argv[1]);
    outFile.open(argv[2]);
    debuggingFile.open(argv[3]);

    int bucketsize = 29;
    HashTable* hashtable = new HashTable(bucketsize);
    string currentData = "";
    int index = 0;

    while(inFile >> currentData){
        index = hashtable->hashString(currentData);
        hashtable->hashInsert(index, currentData, outFile,
debuggingFile);
        hashtable->printList(index, debuggingFile);
    }
    outFile <<endl<<endl<<endl<< "Final print: "<<endl;
    hashtable->printHashTable(outFile);

    inFile.close();
    outFile.close();
    debuggingFile.close();

    return 0;
}

```

OutFile 1:

```
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
*** Data is already in the hashtable, no insertion takes place
```

Final print:

```
HashTable[0]: (dummy, Huihui) -> (Huihui, Khoi) -> (Khoi, Rashad) ->
(Rashad, Seth) -> (Seth, Wei) -> (Wei, Yangfan) -> (Yangfan, Yechiel)
-> (Yechiel, NULL) ->
HashTable[1]: (dummy, Colin) -> (Colin, Denny) -> (Denny, Michael) ->
(Michael, Tenzin) -> (Tenzin, NULL) ->
HashTable[2]: (dummy, Niraj) -> (Niraj, Xihao) -> (Xihao, NULL) ->
HashTable[3]: (dummy, Conghui) -> (Conghui, Evgeniia) -> (Evgeniia,
Han) -> (Han, Laert) -> (Laert, NULL) ->
HashTable[4]: (dummy, Ameer) -> (Ameer, Nahiyen) -> (Nahiyen, NULL) ->
HashTable[5]: (dummy, Heesun) -> (Heesun, Kevin) -> (Kevin, Lin) ->
(Lin, Patricio) -> (Patricio, NULL) ->
HashTable[6]: (dummy, Cesar) -> (Cesar, Marco) -> (Marco, Matthew) ->
(Matthew, NULL) ->
HashTable[7]: (dummy, Peter) -> (Peter, Xing) -> (Xing, NULL) ->
HashTable[8]: (dummy, Angelo) -> (Angelo, Brian) -> (Brian, NULL) ->
HashTable[9]: (dummy, Joel) -> (Joel, Sim) -> (Sim, Taejoon) ->
(Taejoon, NULL) ->
HashTable[10]: (dummy, Jason) -> (Jason, NULL) ->
HashTable[11]: (dummy, Thurein) -> (Thurein, NULL) ->
HashTable[12]: (dummy, Jacob) -> (Jacob, Yinyu) -> (Yinyu, NULL) ->
HashTable[13]: (dummy, Vincenzo) -> (Vincenzo, Yifei) -> (Yifei, NULL)
->
HashTable[14]: (dummy, Ba) -> (Ba, Esteban) -> (Esteban, Jiayu) ->
(Jiayu, Resfred) -> (Resfred, NULL) ->
HashTable[15]: (dummy, Gregory) -> (Gregory, Kelvin) -> (Kelvin, NULL)
->
HashTable[16]: (dummy, Zhiheng) -> (Zhiheng, NULL) ->
HashTable[17]: (dummy, Carlos) -> (Carlos, Kenny) -> (Kenny, Win) ->
(Win, NULL) ->
HashTable[18]: (dummy, Brandon) -> (Brandon, Cheng) -> (Cheng,
Russell) -> (Russell, Seong) -> (Seong, Youyia) -> (Youyia, NULL) ->
HashTable[19]: (dummy, Ben) -> (Ben, Yuhuan) -> (Yuhuan, NULL) ->
```

```

HashTable[20]: (dummy, Jasmin) -> (Jasmin, Jiaxin) -> (Jiaxin, Luis)
-> (Luis, NULL) ->
HashTable[21]: (dummy, Justin) -> (Justin, Qisheng) -> (Qisheng, NULL)
->
HashTable[22]: (dummy, Emanuel) -> (Emanuel, Hishaam) -> (Hishaam,
Jiade) -> (Jiade, Jingshi) -> (Jingshi, Shaxzod) -> (Shaxzod, Tsaiyun)
-> (Tsaiyun, NULL) ->
HashTable[23]: (dummy, Danielle) -> (Danielle, Hou) -> (Hou, Logan) ->
(Logan, Yuan) -> (Yuan, NULL) ->
HashTable[24]: (dummy, Allan) -> (Allan, Chandra) -> (Chandra,
Christopher) -> (Christopher, Joseph) -> (Joseph, Juan) -> (Juan,
Robin) -> (Robin, NULL) ->
HashTable[25]: (dummy, Gildian) -> (Gildian, Zheng) -> (Zheng, NULL)
->
HashTable[26]: (dummy, Jeffrey) -> (Jeffrey, Jianwei) -> (Jianwei,
NULL) ->
HashTable[27]: (dummy, Andres) -> (Andres, Dorjee) -> (Dorjee,
Taeyong) -> (Taeyong, NULL) ->
HashTable[28]: (dummy, Akshar) -> (Akshar, Zhong) -> (Zhong, NULL) ->

```

Debug:

```

***enter hashInsert method ***
HashTable[9]: (dummy, Joel) -> (Joel, NULL) ->
***enter hashInsert method ***
HashTable[20]: (dummy, Luis) -> (Luis, NULL) ->
***enter hashInsert method ***
HashTable[26]: (dummy, Jianwei) -> (Jianwei, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Yechiel) -> (Yechiel, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Tsaiyun) -> (Tsaiyun, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Hishaam) -> (Hishaam, Tsaiyun) -> (Tsaiyun,
NULL) ->
***enter hashInsert method ***
HashTable[14]: (dummy, Esteban) -> (Esteban, NULL) ->
***enter hashInsert method ***
HashTable[5]: (dummy, Kevin) -> (Kevin, NULL) ->
***enter hashInsert method ***
HashTable[6]: (dummy, Matthew) -> (Matthew, NULL) ->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, NULL) ->
***enter hashInsert method ***
HashTable[27]: (dummy, Taeyong) -> (Taeyong, NULL) ->

```

```

***enter hashInsert method ***
HashTable[25]: (dummy, Gildian) -> (Gildian, NULL) ->
***enter hashInsert method ***
HashTable[14]: (dummy, Esteban) -> (Esteban, Jiayu) -> (Jiayu, NULL)
->
***enter hashInsert method ***
HashTable[23]: (dummy, Hou) -> (Hou, NULL) ->
***enter hashInsert method ***
HashTable[4]: (dummy, Ameer) -> (Ameer, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Hishaam) -> (Hishaam, Jiade) -> (Jiade,
Tsaiyun) -> (Tsaiyun, NULL) ->
***enter hashInsert method ***
HashTable[27]: (dummy, Dorjee) -> (Dorjee, Taeyong) -> (Taeyong, NULL)
->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, Russell) -> (Russell,
NULL) ->
***enter hashInsert method ***
HashTable[7]: (dummy, Xing) -> (Xing, NULL) ->
***enter hashInsert method ***
HashTable[28]: (dummy, Akshar) -> (Akshar, NULL) ->
***enter hashInsert method ***
HashTable[3]: (dummy, Evgeniia) -> (Evgeniia, NULL) ->
***enter hashInsert method ***
HashTable[27]: (dummy, Andres) -> (Andres, Dorjee) -> (Dorjee,
Taeyong) -> (Taeyong, NULL) ->
***enter hashInsert method ***
HashTable[6]: (dummy, Marco) -> (Marco, Matthew) -> (Matthew, NULL) ->
***enter hashInsert method ***
HashTable[21]: (dummy, Justin) -> (Justin, NULL) ->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, Russell) -> (Russell,
Seong) -> (Seong, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Robin) -> (Robin, NULL) ->
***enter hashInsert method ***
HashTable[15]: (dummy, Kelvin) -> (Kelvin, NULL) ->
***enter hashInsert method ***
HashTable[16]: (dummy, Zhiheng) -> (Zhiheng, NULL) ->
***enter hashInsert method ***
HashTable[26]: (dummy, Jeffrey) -> (Jeffrey, Jianwei) -> (Jianwei,
NULL) ->
***enter hashInsert method ***
HashTable[1]: (dummy, Tenzin) -> (Tenzin, NULL) ->
***enter hashInsert method ***
HashTable[17]: (dummy, Carlos) -> (Carlos, NULL) ->
***enter hashInsert method ***
HashTable[13]: (dummy, Yifei) -> (Yifei, NULL) ->

```

```

***enter hashInsert method ***
HashTable[12]: (dummy, Yinyu) -> (Yinyu, NULL) ->
***enter hashInsert method ***
HashTable[7]: (dummy, Peter) -> (Peter, Xing) -> (Xing, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Hishaam) -> (Hishaam, Jiade) -> (Jiade,
Shaxzod) -> (Shaxzod, Tsaiyun) -> (Tsaiyun, NULL) ->
***enter hashInsert method ***
HashTable[20]: (dummy, Jiaxin) -> (Jiaxin, Luis) -> (Luis, NULL) ->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, Russell) -> (Russell,
Seong) -> (Seong, Youyia) -> (Youyia, NULL) ->
***enter hashInsert method ***
HashTable[6]: (dummy, Marco) -> (Marco, Matthew) -> (Matthew, NULL) ->
***enter hashInsert method ***
HashTable[13]: (dummy, Vincenzo) -> (Vincenzo, Yifei) -> (Yifei, NULL)
->
***enter hashInsert method ***
HashTable[23]: (dummy, Hou) -> (Hou, Yuan) -> (Yuan, NULL) ->
***enter hashInsert method ***
HashTable[14]: (dummy, Esteban) -> (Esteban, Jiayu) -> (Jiayu,
Resfred) -> (Resfred, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Hishaam) -> (Hishaam, Jiade) -> (Jiade,
Jingshi) -> (Jingshi, Shaxzod) -> (Shaxzod, Tsaiyun) -> (Tsaiyun,
NULL) ->
***enter hashInsert method ***
HashTable[23]: (dummy, Danielle) -> (Danielle, Hou) -> (Hou, Yuan) ->
(Yuan, NULL) ->
***enter hashInsert method ***
HashTable[8]: (dummy, Brian) -> (Brian, NULL) ->
***enter hashInsert method ***
HashTable[10]: (dummy, Jason) -> (Jason, NULL) ->
***enter hashInsert method ***
HashTable[5]: (dummy, Kevin) -> (Kevin, Lin) -> (Lin, NULL) ->
***enter hashInsert method ***
HashTable[25]: (dummy, Gildian) -> (Gildian, Zheng) -> (Zheng, NULL)
->
***enter hashInsert method ***
HashTable[28]: (dummy, Akshar) -> (Akshar, Zhong) -> (Zhong, NULL) ->
***enter hashInsert method ***
HashTable[4]: (dummy, Ameer) -> (Ameer, Nahiyah) -> (Nahiyah, NULL) ->
***enter hashInsert method ***
HashTable[3]: (dummy, Evgeniia) -> (Evgeniia, Han) -> (Han, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Chandra) -> (Chandra, Robin) -> (Robin, NULL)
->
***enter hashInsert method ***

```

```

HashTable[3]: (dummy, Conghui) -> (Conghui, Evgeniia) -> (Evgeniia,
Han) -> (Han, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Chandra) -> (Chandra, Christopher) ->
(Christopher, Robin) -> (Robin, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Chandra) -> (Chandra, Christopher) ->
(Christopher, Juan) -> (Juan, Robin) -> (Robin, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Rashad) -> (Rashad, Yechiel) -> (Yechiel, NULL)
->
***enter hashInsert method ***
HashTable[15]: (dummy, Gregory) -> (Gregory, Kelvin) -> (Kelvin, NULL)
->
***enter hashInsert method ***
HashTable[2]: (dummy, Xihao) -> (Xihao, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Allan) -> (Allan, Chandra) -> (Chandra,
Christopher) -> (Christopher, Juan) -> (Juan, Robin) -> (Robin, NULL)
->
***enter hashInsert method ***
HashTable[19]: (dummy, Yuhuan) -> (Yuhuan, NULL) ->
***enter hashInsert method ***
HashTable[2]: (dummy, Niraj) -> (Niraj, Xihao) -> (Xihao, NULL) ->
***enter hashInsert method ***
HashTable[23]: (dummy, Danielle) -> (Danielle, Hou) -> (Hou, Logan) ->
(Logan, Yuan) -> (Yuan, NULL) ->
***enter hashInsert method ***
HashTable[14]: (dummy, Ba) -> (Ba, Esteban) -> (Esteban, Jiayu) ->
(Jiayu, Resfred) -> (Resfred, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Khoi) -> (Khoi, Rashad) -> (Rashad, Yechiel) ->
(Yechiel, NULL) ->
***enter hashInsert method ***
HashTable[22]: (dummy, Hishaam) -> (Hishaam, Jiade) -> (Jiade,
Jingshi) -> (Jingshi, Shaxzod) -> (Shaxzod, Tsaiyun) -> (Tsaiyun,
NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Khoi) -> (Khoi, Rashad) -> (Rashad, Seth) ->
(Seth, Yechiel) -> (Yechiel, NULL) ->
***enter hashInsert method ***
HashTable[12]: (dummy, Jacob) -> (Jacob, Yinyu) -> (Yinyu, NULL) ->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, Russell) -> (Russell,
Seong) -> (Seong, Youyia) -> (Youyia, NULL) ->
***enter hashInsert method ***
HashTable[17]: (dummy, Carlos) -> (Carlos, Win) -> (Win, NULL) ->
***enter hashInsert method ***
HashTable[11]: (dummy, Thurein) -> (Thurein, NULL) ->

```



```

***enter hashInsert method ***
HashTable[27]: (dummy, Andres) -> (Andres, Dorjee) -> (Dorjee,
Taeyong) -> (Taeyong, NULL) ->
***enter hashInsert method ***
HashTable[5]: (dummy, Kevin) -> (Kevin, Lin) -> (Lin, Patricio) ->
(Patricio, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Allan) -> (Allan, Chandra) -> (Chandra,
Christopher) -> (Christopher, Juan) -> (Juan, Robin) -> (Robin, NULL)
->
***enter hashInsert method ***
HashTable[8]: (dummy, Angelo) -> (Angelo, Brian) -> (Brian, NULL) ->
***enter hashInsert method ***
HashTable[1]: (dummy, Denny) -> (Denny, Tenzin) -> (Tenzin, NULL) ->
***enter hashInsert method ***
HashTable[3]: (dummy, Conghui) -> (Conghui, Evgeniia) -> (Evgeniia,
Han) -> (Han, Laert) -> (Laert, NULL) ->
***enter hashInsert method ***
HashTable[16]: (dummy, Zhiheng) -> (Zhiheng, NULL) ->
***enter hashInsert method ***
HashTable[9]: (dummy, Joel) -> (Joel, Taejoon) -> (Taejoon, NULL) ->
***enter hashInsert method ***
HashTable[5]: (dummy, Heesun) -> (Heesun, Kevin) -> (Kevin, Lin) ->
(Lin, Patricio) -> (Patricio, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Allan) -> (Allan, Chandra) -> (Chandra,
Christopher) -> (Christopher, Joseph) -> (Joseph, Juan) -> (Juan,
Robin) -> (Robin, NULL) ->
***enter hashInsert method ***
HashTable[9]: (dummy, Joel) -> (Joel, Sim) -> (Sim, Taejoon) ->
(Taejoon, NULL) ->
***enter hashInsert method ***
HashTable[17]: (dummy, Carlos) -> (Carlos, Kenny) -> (Kenny, Win) ->
(Win, NULL) ->
***enter hashInsert method ***
HashTable[20]: (dummy, Jasmin) -> (Jasmin, Jiaxin) -> (Jiaxin, Luis)
-> (Luis, NULL) ->
***enter hashInsert method ***
HashTable[19]: (dummy, Ben) -> (Ben, Yuhuan) -> (Yuhuan, NULL) ->
***enter hashInsert method ***
HashTable[21]: (dummy, Justin) -> (Justin, Qisheng) -> (Qisheng, NULL)
->
***enter hashInsert method ***
HashTable[1]: (dummy, Denny) -> (Denny, Michael) -> (Michael, Tenzin)
-> (Tenzin, NULL) ->
***enter hashInsert method ***
HashTable[18]: (dummy, Brandon) -> (Brandon, Cheng) -> (Cheng,
Russell) -> (Russell, Seong) -> (Seong, Youyia) -> (Youyia, NULL) ->
***enter hashInsert method ***

```

```

HashTable[18]: (dummy, Brandon) -> (Brandon, Cheng) -> (Cheng,
Russell) -> (Russell, Seong) -> (Seong, Youyia) -> (Youyia, NULL) ->
***enter hashInsert method ***
HashTable[7]: (dummy, Peter) -> (Peter, Xing) -> (Xing, NULL) ->
***enter hashInsert method ***
HashTable[24]: (dummy, Allan) -> (Allan, Chandra) -> (Chandra,
Christopher) -> (Christopher, Joseph) -> (Joseph, Juan) -> (Juan,
Robin) -> (Robin, NULL) ->
***enter hashInsert method ***
HashTable[8]: (dummy, Angelo) -> (Angelo, Brian) -> (Brian, NULL) ->
***enter hashInsert method ***
HashTable[1]: (dummy, Colin) -> (Colin, Denny) -> (Denny, Michael) ->
(Michael, Tenzin) -> (Tenzin, NULL) ->
***enter hashInsert method ***
HashTable[8]: (dummy, Angelo) -> (Angelo, Brian) -> (Brian, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Khoi) -> (Khoi, Rashad) -> (Rashad, Seth) ->
(Seth, Wei) -> (Wei, Yechiel) -> (Yechiel, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Khoi) -> (Khoi, Rashad) -> (Rashad, Seth) ->
(Seth, Wei) -> (Wei, Yangfan) -> (Yangfan, Yechiel) -> (Yechiel, NULL)
->
***enter hashInsert method ***
HashTable[22]: (dummy, Emanuel) -> (Emanuel, Hishaam) -> (Hishaam,
Jiade) -> (Jiade, Jingshi) -> (Jingshi, Shaxzod) -> (Shaxzod, Tsaiyun)
-> (Tsaiyun, NULL) ->
***enter hashInsert method ***
HashTable[0]: (dummy, Huihui) -> (Huihui, Khoi) -> (Khoi, Rashad) ->
(Rashad, Seth) -> (Seth, Wei) -> (Wei, Yangfan) -> (Yangfan, Yechiel)
-> (Yechiel, NULL) ->
***enter hashInsert method ***
HashTable[6]: (dummy, Cesar) -> (Cesar, Marco) -> (Marco, Matthew) ->
(Matthew, NULL) ->

```