



**Discipline of Computing and Information Technology**  
**Semester 1, 2022 - SENG1120/6120**

## **Assignment 3**

Due using the Canvas Assignment submission facility:  
11:59PM – Sunday June 5<sup>th</sup>, 2022

### **Specification Version 3.0.1**

**3.0.1. Changes:** corrected the Hash Function – changed `get_code ( )` for the correct function call of `get_name ( )` in the hash function code (*page2 para3*).

NOTE: *The important information about submission and code specifics at the end of this assignment specification.*

## **INTRODUCTION**

In lectures, we have discussed the benefits of using binary search trees and hash tables to store information. In this assignment you will implement both and compare their performances in terms of speed of access.

## **ASSIGNMENT TASK**

You are working in a company that analyses populational data. You are required to create both a Binary Search Tree and a Hash Table data structure to store instances of a class `Cities`.

Both data structures should have functions to *add, remove, display, overloaded operators*, amongst others. **The classes MUST be implemented as class templates.**

You will produce a **Binary Search Tree Container** with;

1. a class template `BTNode`, which is then used as a component in the construction of,
2. a class template `BSTree`, which is our container.
3. Both `BSTree` and `BTNode` should be Dynamic.

You will produce a **Hash Table Container** with;

1. a class template `HTable`, which is our container.

You will be provided with a `TreeHashTableDemo` file, a `Cities` class, and a data file (*text file*) with a list of cities and number of inhabitants; your classes are required to interface with them.

The Binary Search Tree contents must be printed using an **inorder traversal**. The Hash Table must store the instances of `Cities` in an array of size 10,000, and the contents are to be printed from position 0 to n-1; *but only for those positions that contain a valid entry*.

The Hash Function to be used in your Hash Table is provided below. You can copy-and-paste that code to your `HTable` class:

```
template <typename value_type>
int HTable<value_type>::hashfun(const value_type& value) {
    int position = 0;
    string temp = value.get_name();

    for (int i=0; i<(int)temp.length(); i++)
    {
        position += (i+1) * (i+1) * temp.at(i);
    }
    return position % TABLE_SIZE;
}
```

A few points:

- As you implement the classes, you will notice that even though `HTable` and `BSTree` are generic class templates, they only work with `Cities`. That is, some public methods in the `HTable` and `BSTree` classes refer directly (and only make sense) with `Cities`.
- That was a design decision for this assignment. We could get around that, but it would make the assignment extremely challenging for most students.
- You will be able to determine the appropriate Public Interface from the demo file – and your `BSTree` and `HTable` data structures will use the same Public Interface (*same public functions and same overloaded operators*). This will mean you may have to use `add( )` (as per the demo file), instead of `put( )` (as per lecture and well known Public Interfaces).
- When you implement the classes, start small and then add more and more functionality as you go. Start by implementing `BSTree` and comment out everything in the demo file and in the makefile that refers to `HTable`. This way, you can test your code as you go. Once `BSTree` is working, you can move on to `HTable`.
- The solution provided in the next page is the result that you should get if your code is correct. *The computational time will differ*, as this was run in my personal computer, but the hash table should generally be faster than the binary search tree.

And because we're sure some of you will want to know or will want to verify your understanding of what you see in the `TreeHashTableDemo` file:

- The demo code reads the name and population of 244 cities from a file and populates the binary search tree.
- Then, it removes and adds some of the elements 10,000 times, and prints some statistics about the process.
- After that, it does the same for the hash table, and the program finishes.

## SENG6120/BONUS TASK

(1.5 marks) For students in SENG6120, there is an extra requirement:

The 244 cities in the input file provided ***do not induce any collisions*** in a hash table with 10,000 positions. If you reduce the size of the array in the hash table to 1,000 positions, however, ***there will be 19 collisions***. Your task is:

- to create a hash table that eliminates the problem of collisions by using buckets and having a generic linked list in each position of the array (you may use your linked list from Assignment 2).
- This way, if two or more elements map to the same position of the array, they are stored as different nodes in the linked list for that position.
- Test your approach by reducing the size of the array to 1,000 and making sure that all 244 elements are stored correctly, and the demo provided still works.

**For SENG1120 students who want to be challenged more, the above requirement becomes a bonus question, also worth 1.5 marks; *however* you can still only score a MAXIMUM of 15.0/15.0.**

## SUBMISSION

Make sure your code works with the files supplied, and **DO NOT** change them. For marking, we will add the main demo file and the `Cities` class to the project and compile everything using the `makefile`, together with your own files. **If it does not compile or run, your mark will be zero (as we are unable to test and validate your implementation).**

Your submission should be made using the Assignments section of the course Blackboard site. **Incorrectly submitted assignments will not be marked.** You should provide the `.h`/`.hpp` and `.cpp` files related to the `HTable`, `BSTree` and `BSTNode` classes (*if you are attempting the Bonus Question or are a 6120 student*, include your `LinkedList` and `Node` classes as well).

If necessary, provide a `readme.txt` file containing instructions or comments for the marker. Each program file should have a proper header comment section including your name, course and student number; and **your code should be properly documented.**

*Remember that your code will conform to C++98 standards, and should compile and run correctly using Cygwin, gcc and the supplied makefile. There should be no segmentation faults or memory leaks during or after the execution of the program.*

Compress all your files into a *single .zip file*, using your **student number** as the filename (*do not use .rar, .7z, .gz, or any other compressed format*). For example, if your student number is **c9876543**, you would name your submission:

**c9876543.zip**

If you have attempted the Bonus Requirement (*or you are a 6120 student*), please include a blank text file in the same folder as your source files, simply called **Bonus.txt** – this is to make it clear to the marker that you are attempting this.

Submit by selecting the **Assignment 3** link that will be found in the **Assignments** section on **Canvas**.

Late submissions are subject to the rules specified in the Course Outline.

**This assignment is worth 15.0 marks of your final result for the course.**

# EXPECTED OUTPUT

```
/home/Ass3-S1-2022

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022
$

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ make clean
rm -rf *.o core

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ make
g++ -c -Wall -c TreeHashTableDemo.cpp
g++ -c -Wall -c Cities.cpp
g++ TreeHashTableDemo.o Cities.o BTreeNode.h BSTree.h HTable.h -o assignment3

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ ./assignment3.exe
=====
BINARY SEARCH TREE
=====

Initial tree: (Abidjan,4980000) (AddisAbaba,3041002) (Ahmedabad,7717000) (Alexandria,4870000) (Algiers,3415811) (Allahabad,5954391) (Amman,4007526) (Anqing,4723000) (Anshan,3645884) (Atlanta,5449398) (Baghdad,6107000) (Baicheng,3669400) (Bangalore,13999000) (Bangkok,17573000) (Baoding,11860000) (Barcelona,4735000) (Bazhou,3283148) (Beijing,19437000) (BelHorizonte,5159000) (Bengbu,3164467) (Berlin,3664088) (Bijie,6686100) (Binzhou,3748474) (Bogota,7743955) (Boston,4688346) (Bozhou,4850657) (Brasilia,5015268) (BuenosAires,16216000) (Busan,3453198) (Cairo,19787000) (Casablanca,4370000) (Changchun,7674439) (Changde,5827200) (Changsha,8154700) (Changzhi,3334565) (Changzhou,5278121) (Chaoyang,3044641) (Chengde,3473201) (Chengdu,11920000) (Chennai,11564000) (Chenzhou,4744500) (Chicago,8604203) (Chifeng,4341245) (Chongqing,8261000) (Chuzhou,3937868) (Dalian,4480000) (Dallas,5743938) (DarEsSalaam,7461000) (Dazhou,5468097) (Delhi,31870000) (Detroit,3506126) (Deyang,3877000) (Dezhou,5568235) (Dhaka,16839000) (Dongguan,8142000) (Faisalabad,3203846) (Foshan,7905700) (Fuyang,8200264) (Fuzhou,4047200) (Ganzhou,8677600) (Guadalajara,5437000) (Guangan,3205476) (Guangzhou,21489000) (Guigang,4409200) (Guilin,5085500) (Guiyang,4881900) (Handan,9549700) (Hangzhou,6713000) (Hanoi,8246600) (Hyderabad,9840000) (Istanbul,15311000) (Jairpur,3073350) (Jakarta,35362000) (Jeddah,3976000) (Jiangsu,4956600) (Jiangsuanchi,4307488) (Jiangmen,4630300) (Jiaozuo,3590700) (Jieyang,6089400) (Jinan,7967400) (Jining,8081905) (Jinzhou,3126463) (Jiujiang,4896800) (Johannesburg,4434827) (Kabul,4273156) (Kano,3848885) (Karachi,15292000) (Khartoum,6017000) (Kinshasa,15056000) (Kolkata,18698000) (KualaLumpur,8639000) (Kunming,6850000) (Lagos,15487000) (Lahore,11148000) (Langfang,4358839) (Lanzhou,3616163) (Leshan,3235759) (Liaocheng,5789863) (Lima,8992000) (Linyi,4316610) (Linyi,10820000) (Liuzhou,4041700) (London,11200000) (LosAngeles,12750807) (Loudi,3931800) (Luan,5611701) (Luanda,8883000) (Lucknow,3382000) (Luoyang,6888500) (Luzhou,4218427) (Madrid,6006000) (Manila,23971000) (Maoming,6313200) (Mashhad,3001184) (Meizhou,4378800) (Melbourne,4529500) (MexicoCity,21505000) (Miami,6445545) (Mianyang,4613862) (Montreal,3519595) (Moscow,17693000) (Mumbai,22186000) (Nagoya,9522000) (Nanchang,5545500) (Nanchong,6278614) (Nangando,5708191) (Nanjing,7299000) (Nanning,7254100) (Nantong,7283622) (Nanyang,10013600) (Neijiang,3702847) (NewYork,18713220) (Osaka,15490000) (Paris,11027000) (Philadelphia,5649300) (Phoenix,4219697) (Pingdingshan,4904701) (Pudong,5187200) (Pune,7948000) (Puyang,3598740) (Qinching,3262549) (Qingdao,5775000) (Qingyuan,3874000) (Qinhuangdao,3146300) (Qinzhou,3304400) (Qiqihar,5367003) (Quanzhou,6480000) (Qijiang,6155400) (Rangoon,5430000) (RiodeJaneiro,12486000) (Riyadh,6889000) (SaintPetersburg,5384342) (SanDiego,3220118) (SanFrancisco,3592294) (SantaCruz,3151676) (Santiago,7026000) (SaoPaulo,722495000) (Seattle,3789215) (Seoul,22394000) (Shanghai,22118000) (Shangqiu,7325300) (Shangrao,6810700) (Shantou,5319028) (Shaoyang,7370500) (Shenyang,7208000) (Shenzhen,14678000) (Shijiazhuang,10784600) (Shiyan,3398000) (Singapore,5271000) (Siping,3385156) (Suihua,5418153) (Suining,3252619) (Surat,4875000) (Suzhou,5352924) (Sydney,4840600) (Taipei,5494207) (Taiyuan,4201592) (Taizhou,5031000) (Tangshan,7577289) (Tehran,13819000) (Tianjin,10932000) (Tibiao,444444) (Tokyo,39105000) (Tongliao,3139153) (Tongren,3168800) (Tongshan,8669000) (Urumqi,3112559) (Washington,5379184) (Weifang,9373000) (Weinan,5286077) (Wuhan,9729000) (Wuhu,3842100) (Wuxi,6372624) (Wuzhou,3061100) (Xi'an,4110000) (Xi'an,7090000) (Xiangyang,5680000) (Xianyang,5096001) (Xiaogan,4921000) (Xiaoxita,4137900) (Xingtai,7104103) (Xinpu,4393482) (Xinyang,6109106) (Xinzhong,3067501) (Yancheng,7260240) (Yangzhou,4459760) (Yantai,6968202) (Yibin,4471896) (Yichun,5573200) (Yiyang,4413800) (Yokohama,3576300) (Yongzhou,5452100) (Yulin,5849700) (Yulinshi,3351436) (Yuncheng,5134779) (Zaozhuan,3729140) (Zhangjiakou,4345485) (Zhanjiang,7332000) (Zhaoqing,4151700) (Zhaotong,5591000) (Zhengzhou,10136000) (Zhenjiang,3113384) (Zhongshan,3260000) (Zhoukou,8953172) (Zhumadian,7231234) (Zhuzhou,4020800) (Zibo,4530597) (Zunyi,6270700)

Adding and removing...

Time elapsed: 1.234 seconds
Time per ins/del operation: 3.98065 milliseconds.
There are 1743915249 people living in those cities and 663128227 people living in cities with more than 10 million inhabitants.

=====
HASH TABLE
=====

Initial hash table: (Sydney,4840600) (Alexandria,4870000) (Boston,4688346) (Zhongshan,3260000) (Moscow,17693000) (Bazhou,3283148) (Dazhou,5468097) (Xiangyang,5680000) (Dezhou,5568235) (Jaipur,3073350) (Bozhou,4850657) (Fuzhou,4047200) (Johannesburg,4434827) (Luzhou,4218427) (Suzhou,5352924) (Wuzhou,3061100) (Bangalore,13999000) (Zaozhuan,3729140) (Chongqing,8261000) (Singapore,5271000) (LosAngeles,12750807) (Changchun,7674439) (Melbourne,4529500) (DarEsSalaam,7461000) (Yokohama,3576300) (Shanghai,22118000) (SanDiego,3220118) (Brasilia,3015268) (Brasilia,3015268) (SantaCruz,3151676) (Nanchang,5545500) (Langfang,4358839) (Changsha,8154700) (Baicheng,3669400) (MexicoCity,21505000) (Neijiang,3702847) (Kinshasa,15056000) (Yancheng,7260240) (Qinching,3262549) (Jiujiang,4896800) (Yuncheng,5134779) (Tongliao,3139153) (Montreal,3519595) (Tangshan,7577289) (Tongshan,8669000) (Santiago,7026000) (Hengyang,7243400) (Jiangmen,4630300) (Nanchong,6278614) (Mianyang,613862) (Chaoyang,3044641) (Xianyang,5096001) (Shaoyang,7370500) (Shenyang,7208000) (Shangrao,6810700) (Dongguan,8142000) (Zhaoqing,4151700) (Changzhou,5278121) (SaoPaulo,722495000) (Guangzhou,21489000) (Zhengzhou,10136000) (Xiaoxita,4137900) (Guadalajara,5437000) (Changzhi,3334565) (Yulinshi,3351436) (Shenzhen,14678000) (Hanzhong,3416196) (Zhaotong,5591000) (Istanbul,15311000) (Qingyuan,3874000) (Hengshui,4472000) (Washington,5379184) (Shangqiu,7325300) (Qinhuangdao,3146300) (Khartoum,6017000) (Guangzhou,6713000) (Yangzhou,4459760) (Chenzhou,4744500) (Yongzhou,5452100) (Lima,8992000) (Jian,4956600) (Xian,7090000) (Pune,7948000) (Zibo,4530597) (Luan,5611701) (Heze,8287693) (Kano,3848885) (BuenosAires,16216000) (Wuxi,6372624) (Wuhu,3842100) (Xiaogan,4921000) (Baogao,6107000) (Mashhad,3001184) (SaintPetersburg,5384342) (Changde,5827200) (Chengde,3473201) (Karachi,15292000) (Weifang,9373000) (Abidjan,4980000) (Chennai,11564000) (Kolkata,18698000) (Guigang,4409200) (Qingdao,5775000) (Chicago,8604203) (Chifeng,4341245) (Guangan,3205476) (Huaihua,4979600) (Huanglongsi,4564900) (Xingtai,7104103) (Baoding,11860000) (Atlanta,5449398) (Beijing,19437000) (Jakarta,35362000) (Jieyang,6089400) (Nanjing,7299000) (Nangando,5708191) (Anshan,3645884) (Zhumadian,7231234) (Huanggang,6333000) (Foshan,7905700) (Linyen,4316610) (Bogota,7743955) (Yiyang,4413800) (Xi'an,4110000) (Jining,8081905) (Tehran,13819000) (Fuyang,8200264) (Harbin,4583000) (Jiading,6155400) (Puyang,3598740) (Suining,3252619) (Anqing,4723000) (Lahore,11148000) (Tibiao,4444444) (Pudong,5187200) (Nagoya,9522000) (Liaocheng,5789863) (Zhanjiang,7332000) (Dallas,5743938) (Chengjiang,3113384) (Shiyan,3398000) (Bengbu,3164467) (Guilin,5085500) (Berlin,3664088) (Barcelona,4735000) (Heyuan,3093900) (London,11200000) (Urumqi,3112559) (Yichun,5573200)

Adding and removing...

Time elapsed: 0.359 seconds
Time per ins/del operation: 1.15806 milliseconds.

There are 1743915249 people living in those cities and 663128227 people living in cities with more than 10 million inhabitants.
The program has finished.

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022
$
```



```
CES236-7DXQJX2+Al ex@CES236-7DXQJX2 /home/Ass3-S1-2022
$
```

```
CES236-7DXQJX2+Al ex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ make clean
rm -rf *.o core
```

```
CES236-7DXQJX2+Al ex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ make
g++ -c -Wall -c TreeHashTableDemo.cpp
g++ -c -Wall -c Cities.cpp
g++ TreeHashTableDemo.o Cities.o BTNode.h BSTree.h HTable.h -o assignment3
```

```
CES236-7DXQJX2+Al ex@CES236-7DXQJX2 /home/Ass3-S1-2022
$ ./assignment3.exe
```

```
=====
BINARY SEARCH TREE
=====
```

```
Initial tree: (Abidjan, 4980000) (AddisAbaba, 3041002) (Ahmedabad, 7717000) (Alexandria, 4870000)
(Algiers, 3415811) (Allahabad, 5954391) (Amman, 4007526) (Anqing, 4723000) (Anshan, 3645884)
(Atlanta, 5449398) (Baghdad, 6107000) (Baicheng, 3669400) (Bangalore, 13999000) (Bangkok, 17573000)
(Baojing, 11860000) (Barcelona, 4735000) (Bazhou, 3283148) (Beijing, 19437000)
(BeloHorizonte, 5159000) (Bengbu, 3164467) (Berlin, 3664088) (Bijie, 6686100) (Binzhou, 3748474)
(Bogota, 7743955) (Boston, 4688346) (Bozhou, 4850657) (Brasilia, 3015268) (BuenosAires, 16216000)
(Busan, 3453198) (Cairo, 19787000) (Casablanca, 4370000) (Changchun, 7674439) (Changde, 5827200)
(Changsha, 8154700) (Changzhi, 3334565) (Changzhou, 5278121) (Chaoyang, 3044641) (Chengde, 3473201)
(Chengdu, 11920000) (Chennai, 11564000) (Chenzhou, 4744500) (Chicago, 8604203) (Chifeng, 4341245)
(Chongqing, 8261000) (Chuzhou, 3937868) (Dalian, 4480000) (Dallas, 5743938) (DarEsSalaam, 7461000)
(Dazhou, 5468097) (Delhi, 31870000) (Detroit, 3506126) (Deyang, 3877000) (Dezhou, 5568235)
(Dhaka, 16839000) (Dongguan, 8142000) (Faisalabad, 3203846) (Foshan, 7905700) (Fuyang, 8200264)
(Fuzhou, 4047200) (Ganzhou, 8677600) (Guadalajara, 5437000) (Guangan, 3205476) (Guangzhou, 21489000)
(Guigang, 4409200) (Guilin, 5085500) (Guiyang, 4881900) (Handan, 9549700) (Hangzhou, 6713000)
(Hanoi, 8246600) (Hanzhong, 3416196) (Harbin, 4583000) (Hechi, 3545700) (Hefei, 7457027)
(Hengshui, 4472000) (Hengyang, 7243400) (Heyuan, 3093900) (Heze, 8287693) (HoChiMinhCity, 13954000)
(HongKong, 7398000) (Houston, 5464251) (Huainan, 4799889) (Huaihua, 4979600) (Huanggang, 6333000)
(Huanglongsi, 4564900) (Huizhou, 4830000) (Hyderabad, 9840000) (Istanbul, 15311000) (Jai pur, 3073350)
(Jakarta, 35362000) (Jeddah, 3976000) (Jian, 4956600) (Jiangguanchi, 4307488) (Jiangmen, 4630300)
(Jiaozuo, 3590700) (Jiayang, 6089400) (Jinan, 7967400) (Jining, 8081905) (Jinzhou, 3126463)
(Jiujiang, 4896800) (Johannesburg, 4434827) (Kabul, 4273156) (Kano, 3848885) (Karachi, 15292000)
(Khartoum, 6017000) (Kinshasa, 15056000) (Kolkata, 18698000) (KualaLumpur, 8639000)
(Kunming, 6850000) (Lagos, 15487000) (Lahore, 11148000) (Langfang, 4358839) (Lanzhou, 3616163)
(Leshan, 3235759) (Liacheng, 5789863) (Lima, 8992000) (Linfen, 4316610) (Linyi, 10820000)
(Liuzhou, 4041700) (London, 1120000) (LosAngeles, 12750807) (Loudi, 3931800) (Luan, 5611701)
(Luanda, 8883000) (Lucknow, 3382000) (Luoyang, 6888500) (Luzhou, 4218427) (Madrid, 6006000)
(Manila, 23971000) (Maoming, 6313200) (Mashhad, 3001184) (Meizhou, 4378800) (Melbourne, 4529500)
(MexicoCity, 21505000) (Miami, 6445545) (Mianyang, 4613862) (Montreal, 3519595) (Moscow, 17693000)
(Mumbai, 22186000) (Nagoya, 9522000) (Nanchang, 5545500) (Nanchong, 6278614) (Nangandao, 5708191)
(Nanjing, 7729000) (Nanning, 7254100) (Nantong, 7283622) (Nanyang, 10013600) (Neijiang, 3702847)
(NewYork, 18713220) (Osaka, 15490000) (Paris, 11027000) (Philadelphia, 5649300) (Phoenix, 4219697)
(Pingdingshan, 4904701) (Pudong, 5187200) (Pune, 7948000) (Puyang, 3598740) (Qincheng, 3262549)
(Qingdao, 5775000) (Qingyuan, 3874000) (Qinhuangdao, 3146300) (Qinzhou, 3304400) (Qiqihar, 5367003)
(Quanzhou, 6480000) (Qujing, 6155400) (Rangoon, 5430000) (RiodeJaneiro, 12486000) (Riyadh, 6889000)
(SaintPetersburg, 5384342) (SanDiego, 3220118) (SanFrancisco, 3592294) (SantaCruz, 3151676)
(Santiago, 7026000) (SaoPaulo, 22495000) (Seattle, 3789215) (Seoul, 22394000) (Shanghai, 22118000)
(Shangqiu, 7325300) (Shangrao, 6810700) (Shantou, 5319028) (Shaoyang, 7370500) (Shenyang, 7208000)
(Shenzhen, 14678000) (Shijiazhuang, 10784600) (Shiyang, 3398000) (Singapore, 5271000)
(Siping, 3385156) (Suihua, 5418153) (Suiyuan, 3252619) (Surat, 4875000) (Suzhou, 5352924)
(Sydney, 4840600) (Tai'an, 5494207) (Taiyuan, 4201592) (Taizhou, 5031000) (Tangshan, 757289)
(Tehran, 13819000) (Tianjin, 10932000) (Timbiso, 4444444) (Tokyo, 39105000) (Tongliao, 3139153)
(Tongren, 3168800) (Tongshan, 8669000) (Urumqi, 3112559) (Washington, 5379184) (Weifang, 9373000)
(Weinan, 5286077) (Wuhan, 9729000) (Wuhu, 3842100) (Wuxi, 6372624) (Wuzhou, 3061100)
(Xiamen, 4110000) (Xian, 7090000) (Xi'angyang, 5680000) (Xi'anyang, 5096001) (Xi'aoganzhan, 4921000)
(Xiaoxitai, 4137900) (Xingtai, 7104103) (Xinpu, 4393482) (Xinyang, 6109106) (Xinzhou, 3067501)
(Yancheng, 7260240) (Yangzhou, 4459760) (Yantai, 6968202) (Yibin, 4471896) (Yichun, 5573200)
(Yiyang, 4413800) (Yokohama, 3757630) (Yongzhou, 5452100) (Yulin, 5849700) (Yulinshi, 3351436)
(Yuncheng, 5134779) (Zaozhuang, 3729140) (Zhangjiakou, 4345485) (Zhanjiang, 7332000)
(Zhaoqi, 4151700) (Zhaotong, 5591000) (Zhengzhou, 10136000) (Zhenjiang, 3113384)
(Zhongshan, 3260000) (Zhoukou, 8953172) (Zhumadian, 7231234) (Zhuzhou, 4020800) (Zibo, 4530597)
(Zunyi, 6270700)
```

Adding and removing...

Time elapsed: 1.234 seconds  
Time per ins/del operation: 3.98065 milliseconds.  
There are 1743915249 people living in those cities and 663128227 people living in cities with more than 10 million inhabitants.

```
=====
HASH TABLE
=====
```

Initial hash table: (Sydney, 4840600) (Alexandria, 4870000) (Boston, 4688346) (Zhongshan, 3260000)  
(Moscow, 17693000) (Bazhou, 3283148) (Dazhou, 5468097) (Xi'angyang, 5680000) (Dezhou, 5568235)

(Jai pur, 3073350) (Bozhou, 4850657) (Fuzhou, 4047200) (Johannesburg, 4434827) (Luzhou, 4218427)  
 (Suzhou, 5352924) (Wuzhou, 3061100) (Bangalore, 13999000) (Zaozhuang, 3729140) (Chongqing, 8261000)  
 (Singapore, 5271000) (LosAngeles, 12750807) (Changchun, 7674439) (Melbourne, 4529500)  
 (DarEsSalaam, 7461000) (Yokohama, 3757630) (Shanghai, 22118000) (SanDiego, 3220118) (HongKong, 7398000)  
 (Brasilia, 3015268) (SantaCruz, 3151676) (Nanchang, 5545500) (Langfang, 4358839) (Changsha, 8154700)  
 (Baicheng, 3669400) (MexicoCity, 21505000) (Neijiang, 3702847) (Kinshasa, 15056000) (Yancheng, 7260240)  
 (Qincheng, 3262549) (Jiujiang, 4896800) (Yuncheng, 5134779) (Tongliao, 3139153) (Montreal, 3519595)  
 (Tangshan, 7577289) (Tongshan, 8669000) (Santiago, 7026000) (Hengyang, 7243400) (Jiangmen, 4630300)  
 (Nanchong, 6278614) (Mianyang, 4613862) (Chaoyang, 3044641) (Xiayang, 5096001) (Shaoyang, 7370500)  
 (Shenyang, 7208000) (Shangrao, 6810700) (Dongguan, 8142000) (Zhaoqing, 4151700) (Changzhou, 5278121)  
 (SaoPaulo, 22495000) (Guangzhou, 21489000) (Zhengzhou, 10136000) (Xiaoxitai, 4137900)  
 (Guadalajara, 5437000) (Changzhi, 3334565) (Yulinshi, 3351436) (Shenzhen, 14678000) (Hanzhong, 3416196)  
 (Zhaotong, 5591000) (Istanbul, 15311000) (Qingyuan, 3874000) (Hengshui, 4472000) (Washington, 5379184)  
 (Shangqiu, 7325300) (Qinhuangdao, 3146300) (Khartoum, 6017000) (Hangzhou, 6713000) (Yangzhou, 4459760)  
 (Chenzhou, 4744500) (Yongzhou, 5452100) (Lima, 8992000) (Jian, 4956600) (Xian, 7090000) (Pune, 7948000)  
 (Zibo, 4530597) (Luan, 5611701) (Heze, 8287693) (Kano, 3848885) (BuenosAires, 16216000) (Wuxi, 6372624)  
 (Wuhu, 3842100) (Xiaganzhan, 4921000) (Baghdad, 6107000) (Mashhad, 3001184) (SaintPetersburg, 5384342)  
 (Changde, 5827200) (Chengde, 3473201) (Karachi, 15292000) (Weifang, 9373000) (Abidjan, 4980000)  
 (Chennai, 11564000) (Kolkata, 18698000) (Guigang, 4409200) (Qingdao, 5775000) (Chicago, 8604203)  
 (Chifeng, 4341245) (Guangan, 3205476) (Huaihua, 4979600) (Huanglongsi, 4564900) (Xingtai, 7104103)  
 (Baoding, 11860000) (Atlanta, 5449398) (Beijing, 19437000) (Jakarta, 35362000) (Jieyang, 6089400)  
 (Nanjing, 7729000) (Zhangjiakou, 4345485) (Nanyang, 10013600) (Maoming, 6313200) (Nanning, 7254100)  
 (Guiyang, 4881900) (Xinyang, 6109106) (Qiqihar, 5367003) (Suining, 3252619) (Kunming, 6850000)  
 (Luoyang, 6888500) (Seattle, 3789215) (Tianjin, 10932000) (Bangkok, 17573000) (Tongren, 3168800)  
 (Chengdu, 11920000) (Nantong, 7283622) (NewYork, 18713220) (Taiyuan, 4201592) (Rangoon, 5430000)  
 (HoChiMinhCity, 13954000) (Algiers, 3415811) (Quanzhou, 6480000) (Dhaka, 16839000) (Phoenix, 4219697)  
 (Osaka, 15490000) (Detroit, 3506126) (KualaLumpur, 8639000) (Hefei, 7457027) (Bijie, 6686100)  
 (Hechi, 3545700) (Houston, 5464251) (Taizhou, 5031000) (Meizhou, 4378800) (Tai'an, 5494207)  
 (Lucknow, 3382000) (Ganzhou, 8677600) (Delhi, 31870000) (Lanzhou, 3616163) (Miami, 6445545)  
 (Binzhou, 3748474) (Shantou, 5319028) (Jinzhou, 3126463) (Huizhou, 4830000) (Qinzhou, 3304400)  
 (Xinzhou, 3067501) (Zhoukou, 8953172) (Amman, 4007526) (Jinan, 7967400) (Wuhan, 9729000) (Loudi, 3931800)  
 (Chuzhou, 3937868) (Yibin, 4471896) (Luzhou, 4041700) (Zhuzhou, 4020800) (Jiaozuo, 3590700)  
 (Hanoi, 8246600) (Busan, 3453198) (Kabul, 4273156) (Yulin, 5849700) (Cairo, 19787000) (Surat, 4875000)  
 (Lagos, 15487000) (Linyi, 10820000) (Paris, 11027000) (Seoul, 22394000) (Zunyi, 6270700) (Tokyo, 39105000)  
 (Xinpu, 4393482) (AddisAbaba, 3041002) (Philadelphia, 5649300) (Jiangguanchi, 4307488)  
 (Ahmedabad, 7717000) (Allahabad, 5954391) (Faisalabad, 3203846) (Hyderabad, 9840000)  
 (RiodeJaneiro, 12486000) (SanFrancisco, 3592294) (Pingdingshan, 4904701) (Casablanca, 4370000)  
 (Jeddah, 3976000) (Luanda, 8883000) (Shijiazhuang, 10784600) (Mumbai, 22186000) (Manila, 23971000)  
 (BeloHorizonte, 5159000) (Riyadh, 6889000) (Madrid, 6006000) (Handan, 9549700) (Huai'an, 4799889)  
 (Dalian, 4480000) (Yantai, 6968202) (Leshan, 3235759) (Deyang, 3877000) (Suihua, 5418153)  
 (Weinan, 5286077) (Nangandao, 5708191) (Anshan, 3645884) (Zhumadian, 7231234) (Huanggang, 6333000)  
 (Foshan, 7905700) (Linfen, 4316610) (Bogota, 7743955) (Yiyang, 4413800) (Xiamen, 4110000)  
 (Jining, 8081905) (Tehran, 13819000) (Fuyang, 8200264) (Harbin, 4583000) (Qujing, 6155400)  
 (Puyang, 3598740) (Siping, 3385156) (Anqing, 4723000) (Lahore, 11148000) (Timbi'o, 4444444)  
 (Pudong, 5187200) (Nagoya, 9522000) (Liacheng, 5789863) (Zhanjiang, 7332000) (Dallas, 5743938)  
 (Zhenjiang, 3113384) (Shiyan, 3398000) (Bengbu, 3164467) (Guilin, 5085500) (Berlin, 3664088)  
 (Barcelona, 4735000) (Heyuan, 3093900) (London, 11120000) (Urumqi, 3112559) (Yichun, 5573200)

Adding and removing...

Time elapsed: 0.359 seconds

Time per ins/del operation: 1.15806 milliseconds.

There are 1743915249 people living in those cities and 663128227 people living in cities with more than 10 million inhabitants.

The program has finished.

CES236-7DXQJX2+Alex@CES236-7DXQJX2 /home/Ass3-S1-2022  
 \$

**Dan and Alex**

2022-05-15

v.3.0.1(2022-05-27)