COMP 2012 Object-Oriented Programming and Data Structures

Lab 8 Hashing

Image by macrovector on Freepik

Overview

In this lab, you are going to implement a dictionary storing a collection of English abbreviations and their corresponding meanings to demonstrate your understanding of a hash table with open addressing for collision handling. You are required to implement functions of `insert`, `remove`, and `find` with the hash table data structure.

You can find the lecture notes on hashing HERE.
You can download the skeleton code in a zip package HERE.

You will be working with 1 template class, 1 class, and other helper functions. Specifically,

- **HashTable template class** implements a hash table with the following member functions: `constructor`, `destructor`, `offset`, `insert`, `remove`, `find`, and `print`.
- **EnglishAbbrev class** has a hash table data member, and the following member functions: `constructor`, `addAbbrev`, `removeAbbrev`, `searchAbbrev`, and `getNumAbbrevs`.
- **HashTableUtility.h** defines two const variables: `TABLE_SIZE` & `R`, two enum types: `Status` and `CollisionResolutionAlgor`, and two hash functions.

Lab Tasks

You have to complete the following **TODO** tasks so that the program can produce the correct results as expected. In particular, you are required to add appropriate lines of code in 6 different places in **HashTable.tpp**.

TODO #1
Implement the `Constructor`:
- It dynamically allocates an array of `Entry` objects in size `table_size`.
- It also initializes all the other data members using the parameters.

TODO #2
Implement the `Destructor`:
- It deallocates all the dynamically allocated memory to avoid memory leaks.

TODO #3
Implement the `offset` function (equivalent to `f` in the lecture notes):
- It returns the `offset` from the home location (i.e., the hash value computed using `hash_func1`) according to the current collision resolution algorithm, `algo`.

TODO #4
Implement the `insert` function:
- It inserts the key-value pair into the hash table, and returns the number of probes for the insertion of the key-value pair.
- If the key-value pair cannot be inserted to the hash table due to exceeding the maximum number of probes, i.e., `table_size * 2`, the return value should be `table_size * 2`.

TODO #5
Implement the `remove` function:
- It performs lazy deletion for the key-value pair, and returns the number of probes to remove the key-value pair.
- If the key-value pair cannot be removed from the hash table due to exceeding the maximum number of probes, i.e., `table_size * 2`, the return value should be `table_size * 2`.

TODO #6
Implement the `find` function:
- It returns the number of probes for the searching of key-value pair.
- If the key-value pair cannot be found due to exceeding the maximum number of probes, i.e., `table_size * 2`, the return value should be `table_size * 2`.

Sample Sessions

The zip package of the skeleton code includes a **Makefile** designed for the Windows platform. However, if you intend to use it on other platforms such as Linux workstations in CS Lab 2, you may need to make certain modifications for it to function properly.

After completing the assigned tasks (**TODO #1 - TODO #6**) successfully, you can create the executable file, **public_test.exe**, by compiling the files using the command `make all`. You may then test the program using the three sample sessions that are provided below.

Sample Session #1 (user enters **1** to select linear probing as the strategy for resolving collisions)

```
Enter testcase number (1: Linear, 2: Quadratic, 3: Double): 1
**** Insert all the abbreviations ****
Insert: RSVP, Probe: 1, Number of Probes: 1
Insert: ASAP, Probe: 3, Number of Probes: 1
Insert: LMK, Probe: 3, Probe: 4, Number of Probes: 2
Insert: BRB, Probe: 9, Number of Probes: 1
Insert: CC, Probe: 4, Probe: 5, Number of Probes: 2
Insert: DOB, Probe: 8, Number of Probes: 1
Insert: ETA, Probe: 3, Probe: 4, Probe: 5, Probe: 6, Number of Probes: 4

*** Search and Print the meaning ***
Probe: 1, Number of Probes: 1
Meaning of RSVP: Please reply
Probe: 4, Probe: 5, Number of Probes: 2
Meaning of CC: Carbon copy
Probe: 3, Number of Probes: 1
Meaning of ASAP: As soon as possible

*** Remove some abbreviations ***
Probe: 9, Number of Probes: 1
Probe: 4, Probe: 5, Number of Probes: 2

*** Search and Print the meaning ***
Probe: 9, Number of Probes: 1
Meaning of BRB: Not found
Probe: 4, Probe: 5, Number of Probes: 2
Meaning of CC: Not found
```

Sample Session #2 (user enters **2** to select quadratic probing as the strategy for resolving collisions)

```
Enter testcase number (1: Linear, 2: Quadratic, 3: Double): 2
**** Insert all the abbreviations ****
Insert: RSVP, Probe: 1, Number of Probes: 1
Insert: ASAP, Probe: 3, Number of Probes: 1
Insert: LMK, Probe: 3, Probe: 4, Number of Probes: 2
Insert: BRB, Probe: 9, Number of Probes: 1
Insert: CC, Probe: 4, Probe: 5, Number of Probes: 2
Insert: DOB, Probe: 8, Number of Probes: 1
Insert: ETA, Probe: 3, Probe: 4, Probe: 7, Number of Probes: 3

*** Search and Print the meaning ***
Probe: 1, Number of Probes: 1
Meaning of RSVP: Please reply
Probe: 4, Probe: 5, Number of Probes: 2
Meaning of CC: Carbon copy
Probe: 3, Number of Probes: 1
Meaning of ASAP: As soon as possible

*** Remove some abbreviations ***
Probe: 9, Number of Probes: 1
Probe: 4, Probe: 5, Number of Probes: 2

*** Search and Print the meaning ***
Probe: 9, Number of Probes: 1
Meaning of BRB: Not found
Probe: 4, Probe: 5, Number of Probes: 2
Meaning of CC: Not found
```

Sample Session #3 (user enters **3** to select double hashing as the strategy for resolving collisions)

```
Enter testcase number (1: Linear, 2: Quadratic, 3: Double): 3
**** Insert all the abbreviations ****
Insert: RSVP, Probe: 1, Number of Probes: 1
Insert: ASAP, Probe: 3, Number of Probes: 1
Insert: LMK, Probe: 3, Probe: 6, Number of Probes: 2
Insert: BRB, Probe: 9, Number of Probes: 1
Insert: CC, Probe: 4, Number of Probes: 1
Insert: DOB, Probe: 8, Number of Probes: 1
Insert: ETA, Probe: 3, Probe: 4, Probe: 5, Number of Probes: 3

*** Search and Print the meaning ***
Probe: 1, Number of Probes: 1
Meaning of RSVP: Please reply
Probe: 4, Number of Probes: 1
Meaning of CC: Carbon copy
Probe: 3, Number of Probes: 1
```

Submission & Deadline

The deadline of submission for all lab sections is **20:00:00 on 9 May 2023 (Tue)**.

Please submit your completed work to ZINC by zipping the following file.

> **`HashTable.tpp`**

ZINC usage instructions can be found here. You can make multiple submissions to ZINC before the deadline. Only the last submission received before the deadline will be graded.

generating the exact expected outputs as demonstrated above. Apart from the public test case, ZINC will also evaluate your

Menu

Page maintained by

Namkiu Chan

Last Modified: 04/28/2023 12:59:37

Homepage

Course Homepage