Topics: Dictionary and Hashing

ADT's based on Set

1. ADT UID
2. Dictionary
3. Priority Queue

Recall: Difference between ADT set and ADT list

Dictionary

- an ADT based on set with operations:
  1. insert(x,A) – adds element x in set A , if $x \notin A$.
  2. delete(x,A) – removes element x in set A, if $x \in A$.
  3. member(x,A) – returns true (or non-zero) if $x \in A$ else returns false (zero) if $x \notin A$.

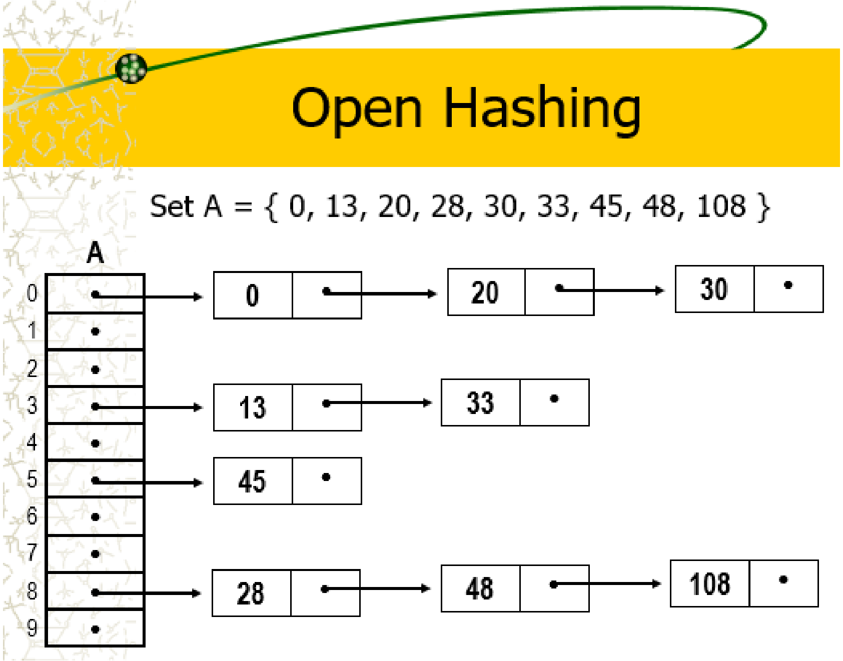
  Utility Functions:

  1. initialize(A) – initializes set A to be empty
  2. makenull(A) – makes set A empty

- Implementations of a Dictionary:
  1. Linked list
  2. Array
  3. Cursor based
  4. Hashing
     - a technique which uses a function called Hash() which determines either of the following:
       1. exact location of the element, OR
       2. the starting location (not necessarily the 1st position) in searching for the element
     - Two types of Hashing
       - A. Open Hashing (or external hashing)
         - This allows the set to be stored in potentially unlimited space.
       - B. Closed Hashing (or internal hashing)
         - This uses a fixed space for storage and thus limits the size of the sets

A] Open Hashing

Illustration:


Open Hashing

Set A = { 0, 13, 20, 28, 30, 33, 45, 48, 108 }

Notes:

1) Open Hashing groups the elements based on a category, via the hash function, i.e. given the element or the key that uniquely identifies the element, the hash function will return the group in which the element belongs.

   - ❖ Example 1: The illustration above groups the integers according to the one's place

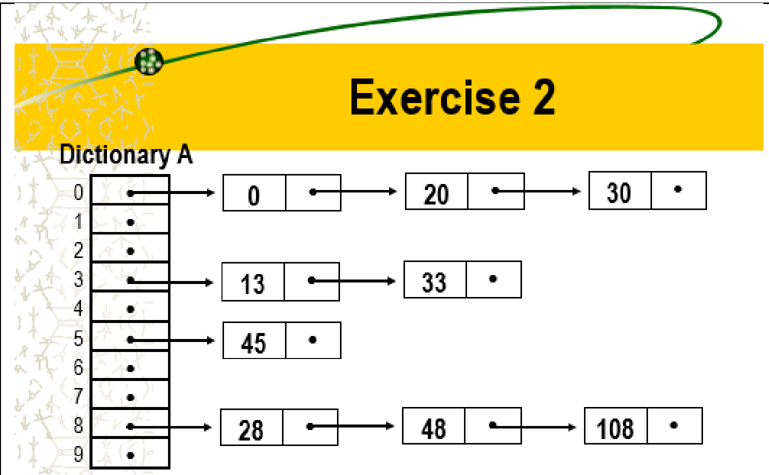
2) If G is the number of groups, the hash function will return a value between 0 to (G-1)

   - ❖ Example: If G = 10, then the hash Value is between 0 to 9 inclusive.

3) The modulo operator is very useful in generating the hash value.

Practice Exercise #1: Write the code of the hash function described in Example 1.

Practice Exercise #2: Dictionary Definition


Exercise 2

Write an appropriate definition of data type Dictionary. Include SIZE as a macro for the size of the array. Note that in the process, a data type of the structure containing the element and the pointer will be defined and a pointer to that structure will also be defined.

Practice Exercise 3: Write a program with the following parts:

A] Include files and macro definition

B] Data Structure definition (see Practice Exercise 2)

C] Function Prototypes of the following functions:

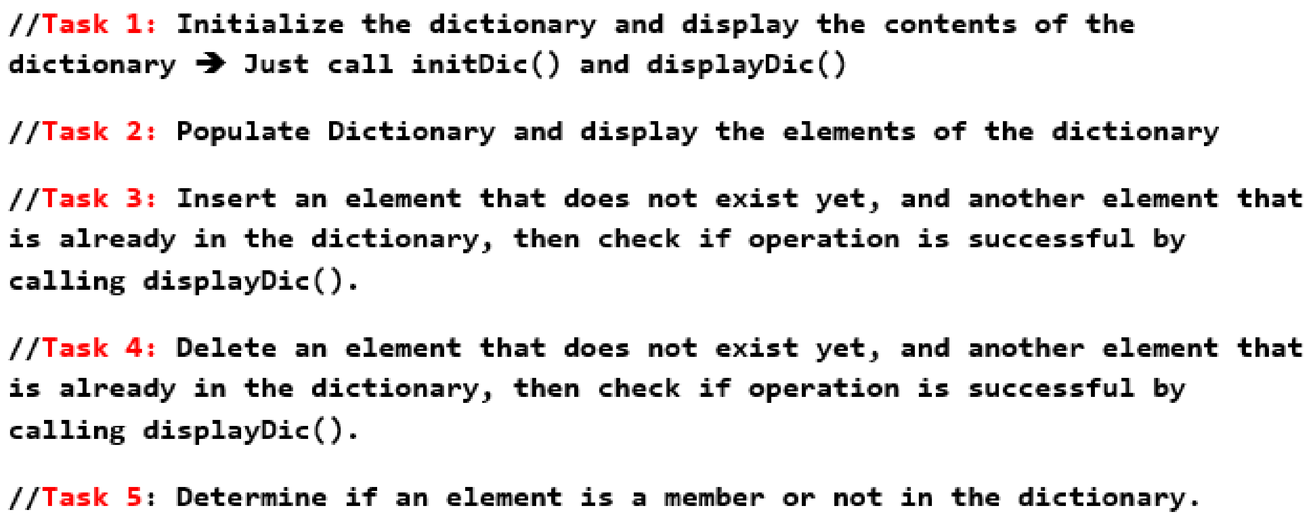
1) hash (see Practice Exercise 1)
2) initDic() - The function will initialize the given Dictionary to be empty.
3) displayDic() – The function will display the Group Number in each line, and for each group display the elements horizontally. Note use %5d instead of \t and you may increase or decrease 5.
4) insert() – Given an element and the dictionary, the function will insert the elements in its proper place in the dictionary if it does not yet exist.
5) populateDic() – Given a set of elements containing, Example: Set A = { 0, 13, 20, 28, 30, 33, 45, 48, 108 }, and a dictionary, the function will call function insert() repeatedly to populate the given dictionary.

6) delete() – Given a dictionary and an element, the function will remove the given element from the dictionary if it exist.

7) isMember() – the function will return TRUE (or 1) if the given element is a member of the given dictionary; otherwise return FALSE (or 0). Note: If TRUE and FALSE will be used define datatype boolean and add it to B] Data Structure definition.

D] Function Main()

Function main contains declaration of variables and function calls:



```
//Task 1: Initialize the dictionary and display the contents of the
dictionary → Just call initDic() and displayDic()

//Task 2: Populate Dictionary and display the elements of the dictionary

//Task 3: Insert an element that does not exist yet, and another element that
is already in the dictionary, then check if operation is successful by
calling displayDic().

//Task 4: Delete an element that does not exist yet, and another element that
is already in the dictionary, then check if operation is successful by
calling displayDic().

//Task 5: Determine if an element is a member or not in the dictionary.
```

E] Write the definition of the functions listed in C]

==

Q: What is the running time of each of the operations insert(), delete(), and isMember()?

Challenge Exercise: Change the given Data structure in Practice exercise 3 to cursor-based implementation.

cfpeña031923