|  |  |
| --- | --- |
| Data Structures & Algorithms Diploma in IT, CSF  Year 2 (2024/25) Semester 4 | Week 6 |
| 2-3 Hours |
| **Practical 6 – Hash Tables** | |

1. Analyze the specification of a Dictionary ADT below. Note that the key type is a string.

|  |
| --- |
| // Dictionary.h - - Specification of Dictionary ADT  #include<string>  #include<iostream>  using namespace std;  const int MAX\_SIZE = 101;  typedef string ItemType;  typedef string KeyType;  struct Node  {  KeyType key; // search key  ItemType item; // data item  Node \*next; // pointer pointing to next item  };  class Dictionary  {  private:  Node \*items[MAX\_SIZE];  int size; // number of items in the Dictionary  public:  // constructor  Dictionary();  // destructor  ~Dictionary();  int hash(KeyType key);  // add a new item with the specified key to the Dictionary  bool add(KeyType newKey, ItemType newItem);  // remove an item with the specified key in the Dictionary  void remove(KeyType key);  // get an item with the specified key in the Dictionary (retrieve)  ItemType get(KeyType key);  // check if the Dictionary is empty  bool isEmpty();  // check the size of the Dictionary  int getLength();    //------------------- Other useful functions -----------------  // display the items in the Dictionary  void print(); print all elements in array  }; |

Implement the operations of the Dictionary ADT

*Note: You should implement (and test) one operation at a time.*

*Hint: You may use the following helper function to get the corresponding integer for a character.*

|  |
| --- |
| int charvalue(char c)  {  if (isalpha(c))  {  if (isupper(c))  return (int)c - (int) 'A';  else  return (int)c - (int) 'a' + 26;  }  else  return -1;  } |

2. Write a sample program, PhoneBookApp.cpp, to do the following:

1. Create an empty dictionary, **phoneBook**.
2. Add the entry, ("Pamela", "64606722"), to the dictionary
3. Add the entry, ("PohSeng", "64608687"), to the dictionary
4. Add the entry, ("Wesley", "64607237"), to the dictionary
5. Add the entry, ("WeeChong", "64606854"), to the dictionary
6. Display all the names in the phone book
7. Add the entry, ("Eugene", "64608256"), to the dictionary
8. Add the entry, ("Saiful", "64608206"), to the dictionary. This entry should collide with ("Eugene", "64608256") Not supposed to have error message.

Use the debugger to look at the data structure that Eugene and Saiful is in a chain.

1. Display all the names in the phone book
2. Add the entry, ("PohSeng", "12345678"), to the dictionary. The program should reject this duplicate entry.
3. Remove the name "Eugene" from the phone book
4. Display all the names in the phone book