Rajalakshmi Engineering College

Name: Rakesh H
Email: 240701415@rajalakshmi.edu.in
Roll no: 240701415
Phone: 7305737702
Branch: REC
Department: I CSE FD
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 7_COD_Question 4

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Develop a program using hashing to manage a fruit contest where each fruit is assigned a unique name and a corresponding score. The program should allow the organizer to input the number of fruits and their names with scores.

Then, it should enable them to check if a specific fruit, identified by its name, is part of the contest. If the fruit is registered, the program should display its score; otherwise, it should indicate that it is not included in the contest.

Input Format

The first line consists of an integer N, representing the number of fruits in the contest.

The following N lines contain a string K and an integer V, separated by a space, representing the name and score of each fruit in the contest.

The last line consists of a string T, representing the name of the fruit to search for.

Output Format

If T exists in the dictionary, print "Key "T" exists in the dictionary.".

If T does not exist in the dictionary, print "Key "T" does not exist in the dictionary.".

Refer to the sample outputs for the formatting specifications.

Sample Test Case

Input: 2
banana 2
apple 1
Banana

Output: Key "Banana" does not exist in the dictionary.

Answer

```
// Simple string hash function
unsigned int hashString(const char* key, int size) {
    unsigned int hash = 0;
    for (int i = 0; key[i] != '\0'; i++) {
        hash += (i + 1) * (unsigned char)key[i]; // Position-weighted sum
    }
    return hash % size;
}

// Check if a key exists in the dictionary using hashing with linear probing
int keyExists(KeyValuePair* dictionary, int size, const char* key) {
    int index = hashString(key, size);
    int original_index = index;

    // Linear probing to find the key
    do {
```

```
        if (strcmp(dictionary[index].key, "") != 0) { // Check if slot is occupied
            if (strcmp(dictionary[index].key, key) == 0) {
                return 1; // Key found
            }
        } else {
            // Empty slot means key doesn't exist
            return 0;
        }
        index = (index + 1) % size; // Move to next slot
    } while (index != original_index);

    return 0; // Key not found (table full or key doesn't exist)
}
```

Status : Correct **Marks : 10/10**