

Rajalakshmi Engineering College

Name: JAGADISH S A

Email: 241501071@rajalakshmi.edu.in

Roll no: 241501071

Phone: 9245831133

Branch: REC

Department: I AI & ML FA

Batch: 2028

Degree: B.E - AI & ML

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

```
#include <iostream>
#include <unordered_map>
#include <string>

using namespace std;

bool keyExists(unordered_map<string, int>& dictionary, string key) {
    return dictionary.find(key) != dictionary.end();
}

int main() {
    int n;
    cin >> n;

    unordered_map<string, int> dictionary;
```

```
for (int i = 0; i < n; ++i) {
    string key;
    int value;
    cin >> key >> value;
    dictionary[key] = value;
}

string key_to_search;
cin >> key_to_search;

if (keyExists(dictionary, key_to_search)) {
    cout << "Key \"<\" exists in the dictionary." << endl;
} else {
    cout << "Key \"<\" does not exist in the dictionary." <<
endl;
}

return 0;
}
```

Status : Correct

Marks : 10/10