

2020-2021 Fall Semester

Algorithm Analysis

Assignment – 2

Course Coordinators

Assoc. Dr. Prof. Dr. M. Elif

KARSLIG YL Instructor Member M. Purpose G UVENSAN

Subject : Using Hashing Algorithm

Problem: In this assignment, a system that lists documents in which a word occurs will be designed. It is very time consuming to look through all the documents at a time to find out which documents a word occurs in. Instead, in this assignment, the words in a newly arrived document will be placed in a dictionary by hashing, and when a word is searched, it will be searched in the dictionary by hashing method and the documents in it will be found.

The system will consist of 3 sub-sections:

1. **Adding the new incoming document to the dictionary:** When a document arrives, pass the words in the document through the hash function, if they are not in the hash table, add both the word and the name of the document to the table, if the word is in the table, add only the name of the document to the address where this word is located.
Store the hash table in a file. When the program is run again, the current file
Add new information using
2. **Searching for the queried word:** When a word is queried, search the word in the table by passing it through the hash function. If there are in the table, print the names of the documents on the screen.
If it is not in the table, indicate with a message that it is not. Print on the screen how many steps the search process is completed.
3. **Creating the hash table:** The hash table will be created according to the following rules:
 - a. *openaddress while* creating the hash table , *double* to solve the conflict problem
hashing methods will be used. According to this:
$$h(\text{key}, i) = [h1(\text{key}) + i \cdot h2(\text{key})] \bmod M$$
$$h1(\text{key}) = \text{key} \bmod M$$
$$h2(\text{key}) = 1 + (\text{key} \bmod MM)$$
 - b. It is assumed that the files contain only words, no special characters.
Words can be mixed with upper and lower case letters. For example, "car" and "car" are the same words. The Horner Method will be used when converting words into numbers.
 - c. Take the **M** value , which represents the table length, as the ASAL number closest to 1000.

 - For the h2 function , take $MM = M - 1$. • Print
the Loadfactor for each newly added word and show a WARNING MESSAGE to the user when the LoadFactor exceeds 0.8. When LoadFactor is 1.0, STOP the ADD process and print the words NOT ADDED in the file to the SCREEN.
 - **TIP: Store the last value of the LoadFactor in the file. newly added
UPDATE this value for words.**

Important information about Homework Submission:

Prepare a single document containing all the information given below and upload the StudentNumarasi.rar file from online.yildiz.edu.tr until **02.12.2020 at 23:59** .

Deliverables:

1. Prepare the program of your algorithm in **C** language and add it to the document.
2. Share the screen outputs showing the different states for each algorithm in the document.
3. In the first question, do the complexity analysis for both options.

4. Deliverables

- a. StudentNumarasi.rar (Ex: 15011001.rar) i.
StudentNumber.pdf (Ex: 15011001.pdf) ii.
StudentNumarasi.c (Ex: 15011001.c) iii. StudentNumber.txt
(The final version of the file where the Hash Table is stored
Example: 15011001.txt)

Evaluation

Algorithm Design and Program Operation: (80%)

1. Assignment must complete all of the required tasks.
2. A design that is free from unnecessary controls and processes should be made.
3. The program should run without errors.
4. During the operation of the program, the introduction and
Outputs should be informed by messages.

Report Documentation: (20%)

1. On the cover page of the report, the name of the course, the student's name, surname and number, and homework information should take place.
2. While declaring a variable in the source code, each variable should be defined on a single line.
It should be written next to it as a description of what the variable will be used for.
3. Variable names must be meaningful.
4. The work done by each function, its parameters and return value should be explained.
5. Where necessary, the operations made in the code are explained with explanation lines.
6. There should be no unnecessary code duplication.
7. The format of the source code should be neat, readable and traceable.
8. **Be sure to attach the SAMPLE OUTPUT SCREENS to the report.**