2020-2021 Fall Semester

Algorithm Analysis

Homework – 3

Course Coordinators

Assoc. Dr. Prof. Dr. M. Elif

KARSLIGÿL Instructor Member M. Purpose GÜVENSAN

Subject : Dynamic Programming

Problem 1: Teams A and B will play until one of them wins n. The probability of team A winning a match is p for each match, and the probability of losing is 1-p. Therefore, there is no possibility of a draw. The probability of A to win the series is P(i,j) if team A needs to win i more games to win the series and team B needs to win j games to win the series.

(30 Points)

a.) **Recurrence** to design algorithms with **dynamic programming** using P(i,j)
write the **relation .**

b.) In a series of 7 matches where team A has a 0.6 probability of winning a match (4 fields win)
Calculate **the probability of team A winning .** (Use the dynamic programming approach.)

c.) The **code** of the algorithm using the dynamic programming approach that solves the specified problem is in C.
Write in your **language .**

Hint: In a series of 7 matches (4 fields win) where the probability of team A winning a match is 0.4
The probability of winning team A is 0.29.

Problem 2: If there are misspelled words in a questioned sentence, a system will be designed to suggest correct words instead of these words. **(70 Points)**

Example:

User: It is coold

Computer: “coold” is not in the dictionary. Did you mean: “cool “ or “cold”

User: cold

Computer: It is cold

Operational Steps: The operations to be followed in the problem are as follows: 1. To prepare the dictionary to be used in the homework , the words belonging to the dictionary in the **smalldictionary.txt** file are placed in the dictionary *table (hash table) .* This process will be done once.

2. Each word in the given sentence is searched in *the dictionary table .* If the word is in *the dictionary table*
   a. **if** the word is true. The operation is completed for that word. b. **If not, the** *wrong word* is searched in the table.
3. The word *is in the wrong word table (hash table)*
   a. **otherwise ,** the distance of the word to each word in the dictionary table *Levenshtein Edit Distance*
   By calculating with the method, the dictionary words with a distance of **1** are recommended to the user first , and if there is no dictionary words with a distance of 1, the dictionary words with a distance of **2** are recommended to the user. Words with greater distance are not recommended. Wrong word and correct word that the user chose from the suggested words

the word is added to the *incorrect word table.*

b. **If there is** an incorrect word table, this incorrect word is found and the correct word that was previously suggested and accepted by the user for this word is suggested to the user.

4. *Dictionary* and *wrong word* hash tables *openaddress will* be created with *double hashing* methods to solve the conflict problem . You can use the functions you wrote in the 2nd assignment while creating and searching tables.

Bonus: Do not calculate the *edit distance* between the given A[n] and B[m] strings until the end, and the distance between the A and B strings is greater than the value **k** (for this assignment, dictionary words with a distance greater than 2 will not be used, so it is k=2). Those who complete the process in the step that **is certain to** happen will receive **an additional 20 points .** For the given strings A[n] and B[m] and the value k, how many steps is sufficient to advance from [0,0] in the main diagonal in the variation matrix? Please explain briefly. **(20 Points)**

Important information about Homework Submission:

Prepare a single document containing all the information given below and download the StudentNumarasi.rar file from **online.yildiz.edu.tr** until **20.12.2020 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. Deliverables a.

HW#_OgrenciNumarasi.rar (Ex: 15011001.rar) i.

StudentNumber_QuestionNo.pdf (Ex: 15011001_1.pdf) ii.

StudentNumber_QuestionNo_Part.c (Ex: 15011001_1.c) iv.

StudentNumber_QuestionNo.pdf (Ex: 15011001_2.pdf) v.

StudentNumber_QuestionNo.pdf (Ex: 15011001_2.c)

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.