

1 Introduction

In this project, you are given a programming question, and you will develop an efficient algorithm and implement it. You can use any of the programming languages you prefer, there is no limitation. However, usage of any libraries except the standard language library are forbidden. Please request permission if you plan to use any of these (if you are using Python, NumPy is allowed).

Please open the [Google Sheet Link](#), and enter your group number, names of the members and the emails of the members to the cell, where your project is listed. Please keep in mind that the sheet document is divided into different sub-sheets. Thus, select the correct sub-sheet with respect to your group number. All of the test cases for each of the projects can be viewed and downloaded from this [Google Drive Link](#).

Please open a GitHub public repository, include all of your members as contributors and add the repository link to the given Google Sheet document. This step is quite important for us to see your progress and has to be done quickly. Therefore, in the README file please keep a list of completed steps, a TO DO list and the results retrieved if there are any. You will also prepare a presentation and present to the TAs. Therefore, you should also create a Google Slides presentation and include its link to the Google Sheet document.

Please email to comp305staff-group@ku.edu.tr, if there is any problem in viewing the drive folder or modifying the document or if you have some troubles with any of the test cases.

2 Presentation Details

Each of the presentations should take ~ 10 minutes and there will be a 5-minute Q&A session afterwards. If a presentation lasts longer than 10 minutes, then it will be interrupted. During the presentation each of the groups should explain and report:

- The algorithm you designed to solve the problem, the choices of the data structures you used and your reasoning.
- The time complexity of your algorithm (and the space complexity if applicable).
- Your run times for each of the test cases.
- Further improvements that can be done as future works.

This project does not expect from you to come up with just one solution and then test only that solution. For each of the problems you can start with some baseline approaches with more complexity and improve the baseline algorithm step by step. Be as creative as possible. Report different approaches you tested and why did you decide on the final algorithm you present. Your grading will be based on your creativity, your cumulative progress and how well did you present your approach. Additionally, there will be more test cases which won't be provided to you. Therefore, to be sure about the correctness of your algorithm, you should create extra test cases. Finally, having an efficient and fast algorithm is also as important as having correct algorithm. Do your best to find the most efficient and fastest solution. Your effort is really important for grading.

3 Deadlines

You can work on your project until the end of *23th May, 2021*. The project presentations will be held between *24th-28th of May, 2021*.

In the following pages, you can see the available project(s):

Hallelujah Empire

In the Hallelujah Empire, the sons take their name from their fathers with an additional uppercase letter at the beginning of their father's name. For instance if the father's name is ALI, the son's name might be MALI, DALI, WALI etc. And MALI's son's name can be KMALI or PMALI etc. Only the first father in the community does not have a prefix on his name.

4.1 Input and Output

First line is the number of people in the community and number of queries. Following lines are the i th line representing the person i 's first letter in their name and it's fathers id. Second line in the following Sample Input 1 is S 0, which says that this person's name starts with S and 0 means it is the first person in the community and does not inherit a prefix from any other. As no other people in the community have S as a prefix in their name, S is the only prefix of the first person in the community. So for the query S, which is fourth query

4.2 Sample Input Output

Sample Input 1	Sample Output 1
10 5	2
S 0	2
Y 1	1
R 2	1
E 3	0
N 4	
E 5	
A 6	
D 7	
Y 7	
R 9	
RY	
E	
N	
S	
AY	

Figure 1: Sample Input and Output File.

Question Given these information about the names of whole community, and a given query; find the number of people who have the query as a prefix in their name. Provide the most efficient algorithm as you can. Your algorithm will be tested with big input size.