Data Structures and Algorithms

Smart City Management System

Course Project Report

School of Computer Science and Engineering
2023-24

Contents

1. Course and Team Details

1.1 Course details

Course Name	Data Structures and Algorithms
Course Code	20ECSC205
Semester	III
Division	D
Year	2023-24
Instructor	Mallikarjun Akki

1.2 Team Details

Si. No.	Roll No.	Name
1.	440	Veeresh S Hiremath
2.	437	Manoj Pandekamat
3.	438	Rahul Pujari

1.3 Report Owner

Roll No.	Name
440	Veeresh S Hiremath

2. Introduction

From references of "STREETS" document[1] shared by our instructor having series of problems faced by the individuals of the country due to the various issues. From which we noticed the lack of efficient transportation, shortest distances with minimum cost, lack tourist railway booking guide, homeless issues in India, problems of slum areas etc.

Some of the emerging problems adding to this is

1) **Corruption**: Corruption is a widespread and deep-rooted problem in India, affecting various sectors such as public administration, judiciary, law enforcement, education, health, and business [4].
2) **Illiteracy**: Illiteracy is another major problem in India, especially among the rural and marginalized populations [2].
3) **Pollution**: Pollution is a serious environmental and health issue in India, affecting the quality of air, water, and soil [3].
4) **Poverty**: Poverty is a persistent and multidimensional problem in India, affecting millions of people across the country. According to the World Bank, about 22% of India's population (more than 270 million people) lived below the national poverty line of $1.90 per day in 2011[5].
5) **Women's safety**: Women's safety is a crucial and urgent issue in India, as women and girls face various forms of violence, harassment, and discrimination in their daily lives. According to the National Crime Records Bureau, a crime against women was reported every 1.7 minutes in India in 2019, and the most common crimes were domestic violence, rape, kidnapping, and dowry deaths [6].
6) **Natural disaster [7].**

3. Problem Statement

3.1 Domain

Me along with my team after analysing the problems decided to choose some problems of the society and tried provide the best possible efficient solution for the problems which we choose. Problems which we choose are lack of efficient transportation due to which a lot's of time in a day will be wasted waiting for the buses or the railway for the destination we wanted to reach. While designing the city Tokyo in Japan had also faced the similar issues of the transportation from one end to another and of the city [8]. But they used various method in order to find the efficient method to design the city and one of the famous experiments was on the slime mould. They obtained the solution from nature itself by the way slime moulds extended its branches in order the get reach to the food. So from this we understand how in nature the organisms use their ability in order to get the shortest paths [9].

Hospital problems, in recent times due to the "CORONA" virus attack we noticed a lot to medical failure due to the data unavailability from each and every hospital, no proper elite emergency handling system etc in our country [10]. A well maintained structured of the data of not only the hospital but each and every thing must be maintained, so that any emergency conditions in the country will be handled with efficient ways. Our neighbouring country China handled this situation more better than our country even though it spread

the virus all over the world [11]. They are the perfect example to know why the importance of the data in this new era. With the availability of the proper infected patient's data and the number of hospital, their beds, o2 cylinders etc and with their advance construction technology they build world class hospitals within the days in order to counter the situation [12].

Last but not the least the problem with the local hotels and the tourist's spots. The tourist spots play a great role in the development of the overall countries' economy and the tourists spots with beautiful restaurant, hotels will strengthen the near by locality and leads to development. Eg : Maldives is known for its beautiful tourist places i.e. islands where they have all luxury to provide and beyond that they have the proper tourist maps and designs which helps the visitor's to plan their trip with efficiency. Also this they know proper method of advertising. For your knowledge comparing with India, tourism generates only 4.6% of total country GDP [13] in India while in Maldives 25% of its total GDP is from tourism [14].

3.2 Module Description

From the problems we planned to divide the problems and individually work on their respective problem statement. I have selected the problem of hospital. As mentioned above the importance of the hospitals in the emergency conditions, I have tried to maintain the data of the places, distance, hospitals, type of hospital, single rating based on the cleanliness, type of advance technology components ,smart way of booking appoints, checking the information about the available beds in total city hospital, sorting the hospital based on rating, searching the hospital based on places, finding nearest hospital, shortest path connecting all hospital for the city etc.

The name of the functionality suggests their work its work in this system. I have used various data structures and algorithms like hashing, heaps, maps, searching nearest hospital(Dijkstra's), finding shortest path that connects all the hospital in city(prims), Kruskal's, sorting algorithms (heap sort), efficient hospital search method, BST for appointments storing and searching based on the time and hospital key etc. With supporting function like deleting the hospital from the data, deleting places, searching places, removing entire hospitals from the city, removing entire city places, displaying the hospitals of the city, places etc.

4. Functionality Selection

Si. No.	Functionality Name	Known	Unknown	Principles applicable	Algorithms	Data Structures

	Name the functionality within the module	What information do you already know about the module? What kind of data you already have? How much of process information is known?	What are the pain points? What information needs to be explored and understood? What are challenges?	What are the supporting principles and design techniques?	List all the algorithms you will use.	What are the supporting data structures?
1	Add places	The information of the places are stored in the file we have to retrieve the places with its keys line by line	The concepts of hash function and the files should be known in order to generate the hash table and retrieve the data from the file to the ram stored through hash.	Create file consisting of data and through program access the file data line by line and fill the hash table.	Traversing the file and hashing the value.	Hashing + Linked list
2	Delete places	After adding the hospital we want to modify the data	We want to check that the places are present in the file and added if yes then delete.	Search for places and delete only from hash table.	Searching.	Hashing + Linked list
3	Search places	Search places based on their place-key	We want very efficient searching on places.	Usage of Hash function and reducing the traversing length.	Hash function for places .	Hashing + Linked list
4	Delete complete places(Delete table)	Empty the hash for adding new data	There should not be any memory leak.	Traverse through table and free nodes.	Hash function for places	Hashing + Linked list
5	Add hospitals	The information of the hospital are stored in the file we have to retrieve the places with its keys line by line.	The concepts of hash function and the files should be known in order to generate the hash table and retrieve the data from the file to the ram stored through hash.	Create file consisting of data and through program access the file data line by line and fill the hash table.	Traversing the file and hashing the value	Hashing
6	Delete hospitals	After adding the hospital based on the places we wanted to delete hospitals at some places	We must check of places and weather that hospital is present in that places or not	Traverse through table and free nodes	Hash function for places	Hashing
7	Delete full-city hospitals	After adding what if we want to delete all hospital at once	We should empty the hash without any memory leak.	Free all nodes	Traversing	Hashing
8	Search hospital	We want to search hospital based on the places with which we hashed	We want very efficient search method to be implemented	Usage of Hash function and reducing the traversing length	Hash function for hospital and checking the place-key is present or not by search places	Hashing
9	Add paths(graphs).	We want to connect places with distances(wt). They are the routes.	The graph vertices depends on the number of places we added in the hash table	Create the graph using the linked list.	Graph algorithms and linked list.	Graph + Linked list
10	Sort hospital.	Sorting the hospital based on the rating.	We first create the new structure of the copy of data from the	First create the heap, and then we use heapify,	Heap sorting	Heap + Hashing

			main hash and then perform the heap sort.	decrease key, extract min.		
11	Hospitals near me.	Searching the nearest hospital by giving your current location.	We need to find the shortest distance for all near by hospital from the location of the place provided.	Heap, Maps, Distance array, extract min, decrease key, swaping, heapify with correct synchronization.	Dijkstras algorithm, with heap + map data-structure (Greedy algorithm)	Graph + Heaps + Linked list.
12	Add appointment	The function name itself suggests that we want to add appintments.	We want to add appointments according to the hospitals and also it should be stored in such a way that the searching also becomes fast.	We first ask the places and hospital for which the appointment is to be booked and we search for the availability in the BST.	Array of BST	List + Array of roots + BST
13	Shortest route connecting all hospital	Searching the efficient route connecting all the hospital in the city	We want to get the shortest path from the source to all the hospitals it can be even via other places also.	Heap, Maps , Parent array, Extract min, Heapify, Decrease key etc.	Prims algorithm, with heap + Map data-structure (Greedy algorithm)	Graph + Heaps + Linked list
14	Add hotels	Add the data of the hotels liked with places from the file to the main memory for storing data	The data in the file should be valid and the places should be added in the hash before the hotels	Create file consisting of data and through program access the file data line by line and fill the hash table	Traversing the file and hashing the value.	Hash with list
15	Remove appointment	Search weather the appointment is booked at the particular time for the particular hospital and check for the place.	If the the condition match and if the appointment is found only then you must delete the appointment.	We first ask the places and hospital for which the appointment is to be removed and we search for the availability in the BST if we get then remove.	Array of BST	List + Array of roots + BST
16	Show government hospitals, private hospitals, show the total bed available.	We need to go on checking the hash for every hospital in the city.	The hash of the places and the hash for the hospital is very necessary for this process	Traversal through the array of linked list.	Array of linked list traversal method	Array of linked list.
17	Search for appointments	If we book the appointment and we want to search weather it booked or not.	It should be implemented which not only serves for its purpose but also for the add appointment and delete appointment are depending on this	Search in the BST	Array of BST.	List + Array of list + BST
18	Search hotels	Search hotels based on their place-key and hotel key.	The search should be efficient and is based on the places.	Usage of Hash function and reducing the traversing length	Hash function for places	Hashing + Linked list

5. Functionality Analysis

1.Functional analysis of the hash Places function :(Covering the first four functionality)

Efficiency Analysis of
—: Hash-Places :—

(i) Add Key :-

(*) we first read the places from the file until the file is empty.

(*) For reading the places from the file the time complexity will be $O(n)$
n- number of places.

(*) For the function Add key we use
  (i) hash function whose time complexity is $O(1)$.
  (ii) Near prime

$$i \rightarrow \frac{\frac{n}{2} - 2}{1}$$

$$i \rightarrow \frac{n-4}{2} = \frac{n}{2} = O(n)$$

we can neglect this time complexity because the nearprime is used the other hash function so the Amortized time complexity of it is $O(1)$.

so, finally After getting hash index we traverse through the list. but the prime number provided by near prime reduce the process of traversal.
we can consider the add key Time complexity as $O(1)$.

(ii) Delete Key :-
i) we use hash function — $O(1)$.
After getting the hashed index we directly go the provided index and traverse. But the traversal is reduced by the Nearprime function as discussed above.
    so final Time complexity = $O(1)$.

(iii) Search-place key:-
(*) For search place, we read the place-key and directly go to the hashTable directly at the index and traverse for place key

(*) Traverse is reduced by nearprime
    so Time complexity = $O(1)$.
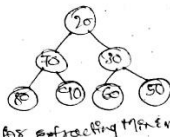
(iv) Delete complete city places :-
(*) For complete deletion of places we go through the every node of the hashtable.
    so final time complexity = $O(n)$
        n- number of places in hash table or Total no of hash keys.
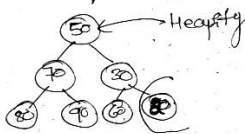
(v) Display the places :-
(*) For displaying the places we traverse all the hash and display

Time complexity = $O(n)$.
        n- number of hash key.

2.Functional analysis of the hash Hospital function :(Covering the next four functionality)

-: Analysis of Hash for Hospitals :-

(i) Add-key i.e AddHospitals :-
(*) For adding the hospital we use file where the data stored in the file will be hospitals according their respective places.
(*) So we retrive all the places line by line from file and add it data to the Hash for the purpose of storing and easy retrival for the data.
(*) For file we need to traverse until hospitals are empty so the time for file is O(n) 'n' is number of hospital in hash.
(*) We use hashfunction for getting index where to store and we directly store at that index.
(*) But due to the usage of linked list we need to traverse but the traverse will be reduced by the nearprime function.
So the Time complexity will be O(1) ignoring the access from file considering that it would be O(n).

(ii) Delete-key :-
(*) For deleting the key we first hash with the given place & hospital key.
(*) then we find/search weather the hospital earth given key is present in the place-key if it satify then only it will add decide to delete key
(*) So the time complexity will be O(1) approximately because the Nearprime will reduce traversal.

Note :- In Add hospital nearprime function is ignored and considering it in the Amortised time complexity

(iii) Search Hospital :-
(*) The main reason for choosing the hash data structure is that its efficiency in the searching of the hospitals.
(*) For searching we need the hospital key and the place key
(*) We use hash function inorder to get the the index and traverse
(*) So the time complexity of the search is O(1).

(iv) For deleting complete hospitals :-
(*) we traverse full hash and delete all the nodes of the hash.
(*) The process should be carried safety so that no memory leak happen.
(*) Time complexity = O(n) where n = number of hospitals in hash table

(v) Display all the Hospital (full city hospital)
(*) It is simial to Above function but change is that here we don't free any of them So, Time complexity = O(n).
n = number of hospitals in hash table

10. Heap sort:

-: Extract Min :-

(*) For extracting Minimum we replace index 0 with index 6

(*) then we reduce the size of heap.

(*) we apply heapify from index (0).

Heapify

* As Already discussed the time complexity for Heapify is $O(\log n)$.

* For interchange the first and last element it takes $O(1)$ time as it is an array.

* so finally the Time complexity of the Extract Min is $O(\log n)$.
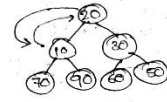   or
   $O(\log places)$.

-: isheap Empty function Analysis :-

(*) Here we just check that the function is consisting of Hash size is zero or not so the time complexity will be $O(1)$.

-: Decrease Key Analysis :-

→ 10
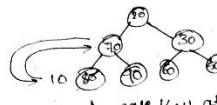
decrease key at index 3 to 10

{ we are going to the parent i.e $\left(\dfrac{i-1}{2}\right)$ for comparing
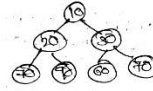
for decreasing at any index it is $O(1)$ but after decreasing we heapify in bottom up manner we are considering only one half. so its similar to case explained earlier.

$$T(n) = T(n/2) + 1$$
$$= T\left(\frac{n}{2^2}\right) + 1 + 1$$
$$= T\left(\frac{n}{2^3}\right) + 1 + 1 + 1$$
$$= \text{for some } k$$
$$= T\left(\frac{n}{2^k}\right) + k(1)$$
$$n = 2^k$$
$$\log n = k \log 2$$
$$k = \frac{\log n}{\log 2}$$
$$k = \log_2 n$$
$$= T(n/n) + \log_2 n (1)$$
$$= 1 + \log_2 n (1)$$
$$= O(\log_2 n).$$

12. Add appointment:

Analysis of Appointment Function :-

(i) Add appointment

(*) The selected data structure for adding the appointment is BST But the problem is that we have different hospital and cannot mix the data of one hospital appointment with another and we also want the efficient searching while removing the appointment

(**) In order to counter this problem we built the Array of BST
i.e
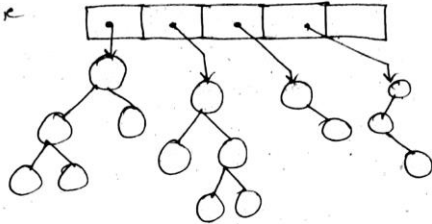
where the indices of array would be representing the hospital key of particular Hospital
Before that we need to search weather the hospital is present in that particular place-key or not
the abovementioned will be done only if the hospital is available at the correct place.

(*) the Binary search tree of a particular hospital stores the appointment patient name and timing and add node based on time.
This will help to reduce the duplicate appointment at the same time in same hospital.

(*) so, finally the time complexity would be
O(logn) becz we compare in BST & either go left or right ignoring the other part.

15. Remove appointment:

(ii) Delete appointment :-

1) For deleting the appointment we first need to check weather the details provided hospital is having that appointment or not.

2) In the BST search for time and if it is found remove it.

3) For removing we need to find inorder successor of the perticular node and re)

4) for removing there are 3-case

(i) Removing of leaf nodes: the time complexity is $O(\log_2 n)$.

(ii) Removing of the partial node :- For them remove node and make parent of removed node directly connecting to the children.
the time complexity is $O(\log_2 n)$.

(iii) Removing the full node :- For removing the full node we find its inorder successor and replace it with the node to be deleated and after replacing it acts like the leaf node and we delete it
So the Time complexity is $O(\log_2 n)$.

so adding $\log_2 n + \log_2 n + \log_2 n = 3\log_2 n$
which is $O(\log_2 n)$.

(iii) display appointment :- (sort Based on Timing virtually)

1) we go the hospital idex and display the BST we used inorder traversal so that the display appointment will sort the appoints by timing and display the appointment.
Time complexity will be $O(n)$.
n – mention here is appointments in all hospital.

16. Government hospital/ Private hospital and total beds analysis:

Analysis of Government and private hospital and bed availability function.

(i) Government / Private hospital ?:-

(*) We traverse whole hash table, i.e every node for checking weather it is a government or private hospital.

(*) The hashing will automatically keep track of the number of elements in the hash table at the particular time

so,
The total time complexity will be
$$i \rightarrow (n-1)$$
where 'n' is the total number of hospitals present in the hash table.
so $O(n)$.

(ii) Bed availability :-

(*) Here also as mentioned above we traverse and get the details of bed availability for each & every hospital so
    Total time complexity = $O(n)$.

12. Nearest Hospital:

Efficiency Analysis of Dijkstra's Algorithm :-

Ex:-
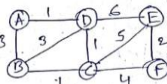
Heap + Map data structure

distance Array

start :- src = A
now decrease key wt of A to 0.
now until the heap+Map ds is uempty extract Min

cur=A & finalize wt in distance array
wt=0
see for Neighbours of A

eg. D ← A check wt+Prwt < ∞ True - decrease the change
    0+1=1< ∞   key - weight in Heap+Map DS
B ← A check 0+3 < ∞ - True - decrease key

Neighbours finished for A so
Again extractMin cur=D & finalize wt in distance array
wt=1
see for Neighbours for D

eg. B ← D check 1+3 < 3 - false - skip
    C ← D check 1+1 < ∞ - True - decrease key
    E ← D check 1+6 < ∞ - True - decrease key
    ExtractMin cur = C & finalize wt.
    wt= 2

eg. B ← C check 2+1 < 3 - false - skip
    F ← C check 2+5 < 7 - false - skip
    F ← C check 2+4 < ∞ - True - decrease key
    ExtractMin cur = B wt=3 & finalize wt

ExtractMin cur=F & Finalize distance
         wt=6
ExtractMin cur=F & Finalize distance
         wt=7

-: Efficiency Analysis :-

We are mainly performing 3 operation
(1) Extract Min — $O(\log n)$.
(2) Heapify — $O(\log n)$
(3) Decrease Key — $O(\log n)$.

we are performing Extract Min until heap is empty.
we know that size of heap = n° of vertices
   so   Time complexity for Extract Min = $O(\log v)$

we are performing decrease key for every extract Min
if the vertex is fully connected then we decrease
it by Edges n° of times so
   Total Time complexity = $E \log(v)$.
                         or
                    $E \log(\text{places})$.

Space complexity = $O(n)$ — for Heap array
                              for Map array
                              for distance array.

14. Shortest route connecting all hospital:

6. Conclusion

From the above analysis we can conclude that these data structures and algorithms on the set of data on hospital is efficient. From this we practically came to know how the real-life problems can be solved using the algorithms, natural ways, through coding etc. I learnt how the problem is to be analysed and different views on the single problem statement will helps us to understand the real-world problems. The main learning is that we implemented algorithms what we have learnt and implemented in one or the other way to solve the problems of the society.

7. References

[1] Streets pdf, Author Sri. Prakesh Hegde.
[2] According to the 2011 census, the literacy rate in India was 74.04%, which means that about 26% of the population (more than 300 million people) were illiterate, Illiteracy limits the opportunities and potential of individuals and communities, and affects their health, income, and social status

[3] India is home to 22 of the world's 30 most polluted cities, according to a 2020 report by IQAir, Water pollution is also a major concern, as more than 70% of India's surface water resources are contaminated by sewage, industrial waste, and agricultural runoff

[4] . Corruption erodes public trust, undermines democracy, hampers development, and fuels inequality.

[5] MPI-22_NITI-Aayog20254.pdf.

[6] [Burning Issue] Women's Safety and India - Civilsdaily.

[7] 31 Causes, Effects & Solutions for Natural Disasters - E&C (environmental-conscience.com).

[8] Slime Mold Grows Network Just Like Tokyo Rail System | WIRED, Ride the Slime Mold Express! | Science | AAAS.

[9] Slime mould attacks simulates Tokyo rail network (nationalgeographic.com).

[10] The human costs of COVID-19 policy failures in India | Nature Human Behaviour.

[11] China, India Handled COVID-19 Differently. Results Differed Too : NPR.

[12] China's coronavirus hospital built in 10 days opens its doors, state media says (nbcnews.com), A construction expert broke down how China built an emergency hospital to treat Wuhan coronavirus patients in just 10 days (businessinsider.in).

[13] Tourism in India - Wikipedia, India: contribution of travel and tourism to GDP 2021 | Statista.

[14] India a Key Player in Maldives' Economic Growth: From Tourism, Hospitality to Infra & Human Resources - News18, How Much Maldives Backlash Could Actually Cost The Tourism-Reliant Island Nation - Benzinga

~*~*~*~*~*~*~