

**SIX WEEKS SUMMER TRAINING**

**REPORT**

on

**DATA STRUCTURE**

Submitted by

**Kartikey Shaurya**

**Registration No 11902836**

**Program Name :- CSE**

Under the Guidance of

**Geeks for Geeks**

**School of Computer Science & Engineering**

**Lovely Professional University**

**DECLARATION**

I hereby declare that I have completed my six weeks summer training at Geeks for Geeks from 1/6/2021 to 15/6/2021 under the guidance of Sandeep Jain. I have declare that I have worked with full dedication during these six weeks of training and my learning outcomes fulfill the requirements of training for the award ofdegree of computer science, Lovely Professional University, Phagwara.

Kartikey shaurya

Date : 15/07/2021 Registration No : 11902836

**ACKNOWLEDGEMENT**

I would like to express my special appreciation and thanks to my Professor Dr. Mr. Sandeep Jain, you have been a tremendous mentor for me. I would like to thank you for encouraging to read Data Structure in depth .Your advice on both reading as well as on my career have been invaluable. I would also like to thank my friend Cheerag Mishra. And also A special thanks to my family. Also I would like to point out , A special thanks to Lovely Professional University for giving me this chance to work on the Data Structures.

Kartikey Shaurya

16/7/2021



**Table of Content**

1. Introduction 5-6

2. Technology Learnt 7-12

3. Reason for choosing this technology. 13

4. Profile of the Problem 14

5. Existing System 15

6. Problem Analysis

• Product definition 16

• Feasibility Analysis 17

7. Software Requirement Analysis 18-19

8. Design

• Flowcharts/Pseudo code 21

9. Implementation 22

10. Learning Outcome from training/technology learnt 23

11. Gantt chart 24

13. Bibliography 25

Times New Roman

**Introduction**

**What are Data Structures ?**

Data Structure can be defined as the collection of data objects which provides a way of storing and managing the data in thhe computer so that it can be used. There are various types of Data Structure such as arrays, linked list , stacks , queues , trees etc. Data Structures are widely used in almost every aspect of Computer Science for simple as well as complex computations. Data structures are used in all areas of computer science such as Artificial Intelligence, graphics, Operating system etc.

Data Structures are the key part of many computer algorithms as they allow the programmers to do data management in an efficient way. A right selection of data structure can enhance the efficiency of computer program or algorithm in a better way.

## Why Data Structures are needed:

With increasing complexities in computer algorithms, the amount of data usage is increasing, this can affect the performance of the application and can create some areas of concern:

1.0 Processing speed:

To handle very large data, high-speed processing is required, but with growing data processor may fail to required processing speed.

2.0 Data Search:

Getting a particular record from should be quick and with optimum use of resources.

3.0 Multiple requests:To handle requests from multiple users

In order to work on concern areas, data structures are used. Data is organized to form a data structure in such a way that all items are not required to be searched and required data can be searched instantly.

## Data Structure Advantages

1.0 Efficient Memory use:With efficient use of data structure memory usage can be optimized, for we can use linked list vs arrays when we are not sure about the size of data. When there is no more use of memory, it can be released.

2.0 Reusability:Data structures can be reused, i.e. once we have implemented a particular data structure, we can use it at any other place. Implementation of data structures can be compiled into libraries which can be used by different clients.

3.0 Abstraction: Data structure serves as the basis of abstract data types, the data structure defines the physical form of ADT(Abstract Data Type). ADT is theoretical and Data structure gives physical form to them.

**Technologies Learnt**

Data Structure is a huge topic but due to limitation of time i have completed these topics and understand deeply how these topics work behind the simple application , which we use

1. Basic Mathematics
2. Bit Magic
3. Recursion
4. Arrays
5. Searching
6. Sorting
7. Matrix
8. Hashing
9. Strings
10. Linked List
11. Stacks
12. Queues
13. Deques

**1. Basic Mathematics**

In this topics there was nothing interesting but all the fundamental mathematics such as lcm, hcf, factorial etc . These topics seems simple but they were used throughout the course

**2. Bit Magics**

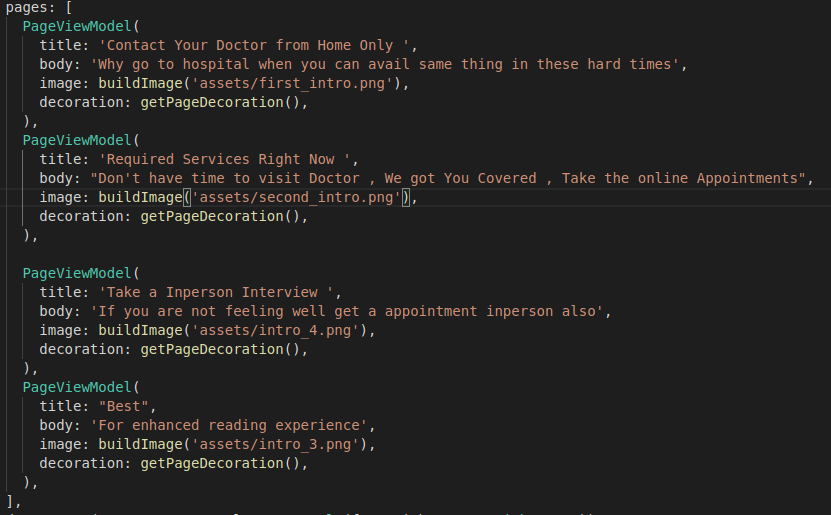
In this particular topic , we find the a powerful use of binary operators and how they can be manipulated to do multiple operations that is much time and memory efficient . Some of the example are Square , shifting the bits and etc.

**3. Recursion**

Recursion , in starting it seems easy but throughout the course this was the most frequent tools that i have used . Also in this topic we have learnt about a interseting problem known as tower of hanaoi.

**4.Arrays**

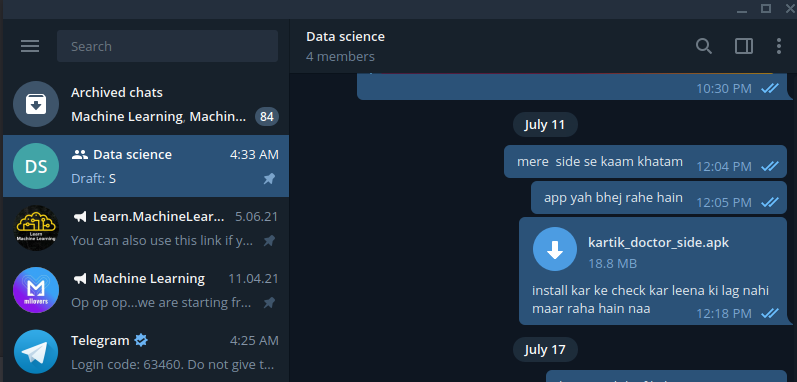
An array is a collection of items of the same data type stored at contiguous memory locations.and probably the most used data structure among all of the data structure. We learned in this topics how the memory location works, and operations on it .In operation Section we learnt about insertion, deletion , array rotation, reversing of array , Sliding window Techniques and Stl use of array.Apart from here i am adding the snapshot of a application that i wrote how this data structure is being used in my application.



This is not a standard array , rather than it is list which behave like array

**5. Searching**

Searching is the key application that we use in our day too day life for example in whatsapp for finding contact we use search techniques . And in this lesson we have seen all the searching techniques how they work and how can we implement it. Some of the technique that we have seen are Binary search , linear search etc

****

**6. Sorting**

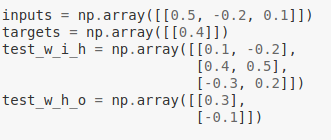
similar to searching sorting is also used heavily in any apllication , and thats why the main project is based on the soring alogorithm for the summer term .

**7. Matrix**

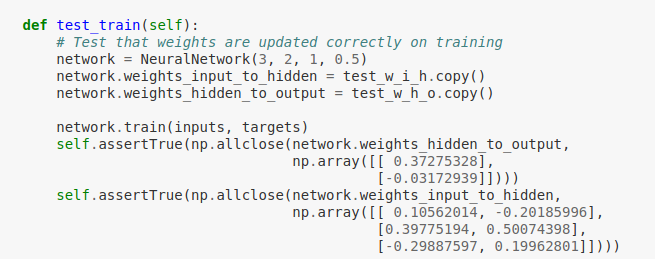
Matrix are also heavly used in the field of ai , basically every calculation is done with the help of matrix and arrays .

In this topic we learnt about different operation related to matrix , some of the examples are multiplication , addition , id etc.

Here is the snap shot how heavily they are used in ML



more of this :



the application are not limited here only , the computations can be made more effective using Dynamic Programming.

**8.Hashing**

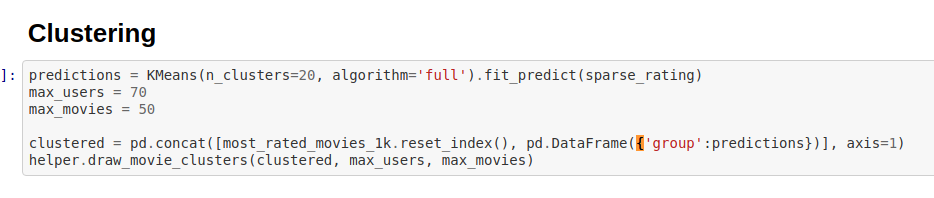
Hashing is also used heavily like any data structure , particularly it is used for storing the data respect to some id here are some quick examples that i have seen in practical example :

1. Insert a phone number and the corresponding information.

2. Search a phone number and fetch the information.

3.Delete a phone number and the related information.

Also this data stucture is known as Dictionary in the python and also it is used heavily in ML for example here is a snapshot from my ml code:-

**9. Strings**

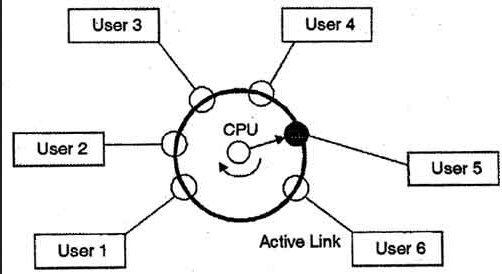
similar to any data structure strings are also used heavily and here is a sample from my code and particulary through this course i have some operations that was pre built in python



all the words that are double quoted

**10 Linked List**

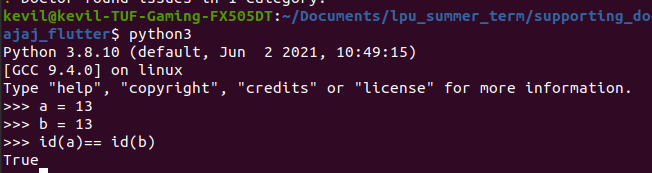
linked list is another great tools for applying where we have time constraint. It is Basically very useful in application of Os for example using the circular linked list we can create a process management tool, where each task is queued in the circular way and when one process successed we can simply drop down it.



Source :- google

**11 Stacks**

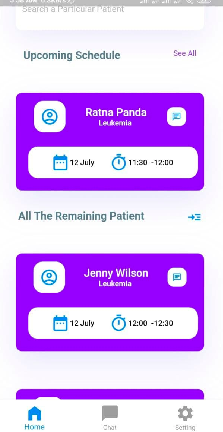
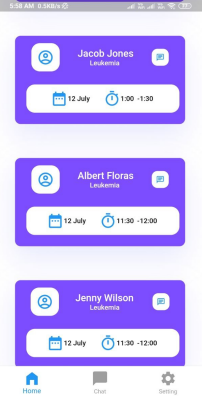
stacks are also important term , it enables the memory management in python , which automatic. And here is a quick example of power of stack , how two variable are managed if they have same Value



here i have just shown that if two varible is having same value then how it is managed

**12 Queues**

Queues are the most interesting data sctructure among all , the simplest example that we can see is instagram stories, tiktok scrolling features and nearly every application uses this particular feature for scrolling here is a quick snap that i have used in my application.



**13. Deques**

Similar to queue the deque is having a tremoundous use , here is a quick example

Think of browser history. You close tabs and if you press Control + Shift + T, the tab, you closed are reopened.Above scenario can be implemented using stack. Last In - First Out. But when this list of browsing history becomes too big in size, it must be able to delete the first entered record, then second and so on as per the requirement.This requires inserting and deleting functionality at both the ends of the queue. This is not the greatest example (I am still searching for one) but I hope it conveys the point.

**Reason for choosing this technology.**

Data structure are used heavly in every field , even in the machine learning , earlier i thought that data structure is not important for my particular field ,but i was wrong . For Example using using wrong data strucutre can increase the time of processing . Let me just give an example :

Suppose we want to make an face prediction app , where we need to have a image and label along with it , for giving the value to machine we can use two approach ,

1. we can use iteration method meaning we can iterate over each image and give label each of it that will take around n\*n for each operation . (like reading or making custom arrys for data preprocessing )

2. Or we can just use dictionary to label the value and the complexity of this will be n around for each operation

mainly there is little difference in speed when we have 200-300 mb of data , but for training around 20 gb of data is very time consuming and resource consuming by first method , hence for understanding such type of problem and their solution i choose this particular topic.

**Profile of the Problem**

The problem statement :-

“

suppose we are given n number of student and the student are preparing for a military parade , and for that we need them to arrange in assending or decending order so that the parade looks good

or other example is

let’s say we are working at google, and we are making a product of google drive such that we are given n number of coping task , and our job is to copying on the basis of size. For example if the size is bigger then it should be coppied first , design an alogithm that can do that .

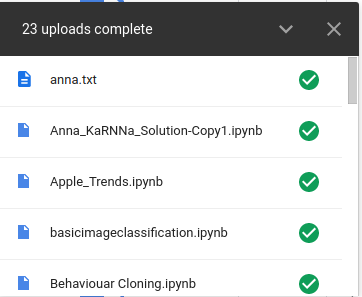
”

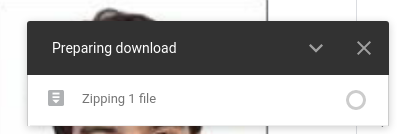
Profiling :

Both of the problem is kinda of either assending or decending so this problem is sorting problem and it can easily approachable using the bubble sort.

**Existing System**

If we look at the google drive then the upload basis is based on the first come first serve basis , hence they are not applying any sorting task , rather than they are applying parallel processing , which means there is no sequence based uploading.

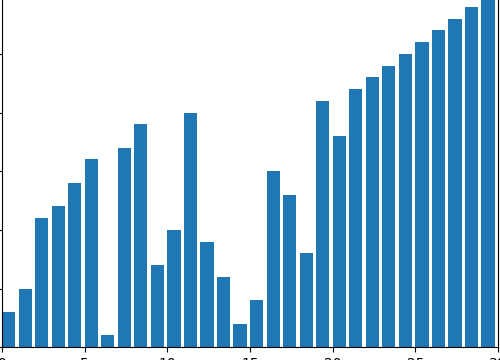


For downloading part they first zip all the file that we select and then download it here is the snapshot of that Problem Analysis

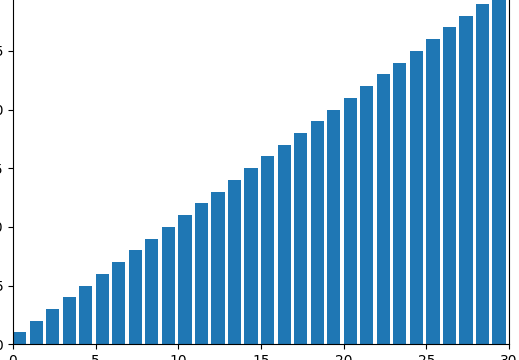
**Product definition**

Here in this project i want to propose a simple algorithm , basically visulize that particular algorithm , that can sort the files on the basis of their size and further that can be implemented in each major software that either upload or download the data.

Intial state the size of the file is basically randoly distributed :-



In the final State the file is sorted and on the basis of internet connectivety the file is downloaded , if the internet connection is good download the biggest size and if the internet connection is slow start with the smallest one

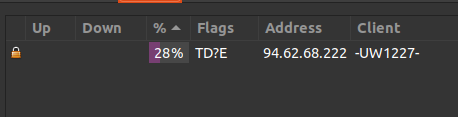


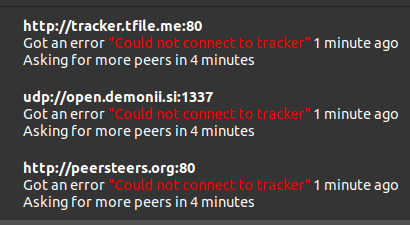
**Feasibility Analysis**

this particular application can impact how we download the data and also can make the consumer much happier.

Apart from this in the torret downloading system we can apply this particular algorithm ,such that the packets that contain the data can be download easily and it will automatically move the traffic to the highest rate uploader and leave the slow connection for the slow downloaders.

Here my connection is connected to the one server but it the file contain on multiple server then , through my algorithm we may have option to connect to fastest or moderate server provider.





**Software Analysis Requirements**

For this Project we mainly required python , matplotlib and seaborn.

A quick word on **Python**

Python is a interpreted high-level general-purpose programming language . The design philosophy emphasizes code readibility and help programmer write clear, logical code for small and large-scale projects.

A quickword on **MATPLOT**

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.It is the oldest library that shows what we need to visualise in the data.

A quickword on  **SEABORN**

Seaborn is a simple and Modern comprehensive library , built on top of the tensorflow to interact , visualize in python .The Only Difference between Matplotlib and Seaborn is Matplot feels old , But functionality wise they both are of same Label.

**DESIGN and PSEUDOCODE**

What is bubble sort ?

Suppose we have a array with these values and we want to sort these

12, 2,14,15,3, 4

the Bubble sort will work like this

1. First it will compare first two , here 12, 2 out of which the smaller it will come in the first index like this

2,12, 14,15,3,4

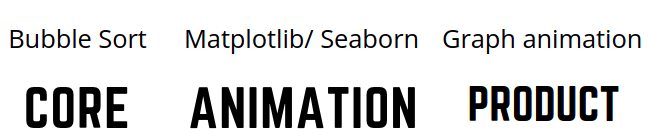
2. Then follwing same method , it will compare 12, 14 and result will look like this

2,12, 14,15,3,4

following same , the algorithm will give result will produce

2,3,4,12,14,15

**DESIGN**



1. Define the function bubble sort

2. Visualize it with the help of MatPLOTLIB and SEABORN

2.1 generate the random value

2.2 Create a figure and subsquent plot

2.3 Set the maximum limit for the X axis

2.4 Update the Value as sorting Proceeds

2.5 finally create and show the animations

2.6 close the plot on pressing ESC

3. Try to implement in a real world application using python .

**PSEUDOCODE**

1. Bubble Sort : the main algorithm

for i in range(n):

for j in range(0, n-i-1):

if arr[j], arr[j+1] = arr[j+1] , arr[j]

2. Making a visualization tools

N = 30

A = list(range(1, N + 1))

random.shuffle(A)

generator = bubblesort(A)

# creates a figure and subsequent subplots

fig, ax = plt.subplots()

ax.set\_title("Bubble Sort ")

bar\_sub = ax.bar(range(len(A)), A, align="edge")

# sets the maximum limit for the x-axis

ax.set\_xlim(0, N)

text = ax.text(0.02, 0.95, "", transform=ax.transAxes)

iteration = [0]

# updating each value in animation

def update(A, rects, iteration):

for rect, val in zip(rects, A):

rect.set\_height(val)

iteration[0] += 1

text.set\_text(f"# of operations: {iteration[0]}")

#2,1 creating the animation for each iteration

anim = animation.FuncAnimation(

fig,

func=update,

fargs=(bar\_sub, iteration),

frames=generator,

repeat=True,

blit=False,

interval=15,

save\_count=90000,

)