# DSA Project Guidelines

# Why to do this project ??

It is a good start to implement what you have learned, by trying to solve a challenging problem

Enhance your problem solving skills

You will develop a solid problem solving approach (will cover that later...)

Don’t do it for the sake of the course project !

# Some basic rules

Programming Language to use – C

Usage of Data Structures to solve the problem

It is a group project

# Why C Programming Language ??

It teaches you to think

Dealing with pointers

Efficiency of the solution

Insight for further projects, programming

# Why Data Structures ?

Capable of solving almost ANY problem

Easy to interpret, visualise, draw

Not too difficult, not too easy

You have this project in your DSA course :)

# How To Find Problem Statement

Choose your domain of interest

Learn all Data-Structures beforehand.
Refer: Abdul Bari lec available in youtube,udemy.

Read various research papers of that subject(domain)

Refer online resources, articles, other people’s algo, approaches
Refer to faculties and set weekly goals.

# Web Design

## DOM Manipulation Tool

Create a command-line tool in C that parses and manipulates a simple HTML document. Use stacks and queues to manage the DOM elements.

## HTML Validator

Implement an HTML validator using a stack to check for matching opening and closing tags.

# Probability and Statistics

Simple Linear Regression:

Implement a linear regression model using arrays to store data points and calculate the best fit line.

K-Nearest Neighbors (KNN) Classifier:

Use linked lists to store data points and implement the KNN algorithm for classification.

Decision Tree Classifier:

Build a decision tree from scratch using tree data structures for classification tasks.

# Spam and Email Analyzer

Implement a spam filter that classifies emails as spam or not based on probability calculations.

These are many more examples.

You can look into field of BioEngineering, AI, LLM, NLP,etc..

We have uploaded detailed list of projects you can think of in the github repo :)

We did our project on domain of NLP and AI.

# Define Problem Statement

After selecting the problem statement, define it!

Decide what you have to implement and what not, depending on its feasibility.

# Deciding Problem Statement

Virtual Piano with Note Playback

Description:

Build a virtual piano that allows users to play notes using a keyboard or mouse, and plays back the corresponding sounds.

Find out what Data Structures you should use :)

# Virtual Piano

Scope of Implementation:
1) Create a graphical user interface (GUI) representing a piano keyboard.
2) Integrate a sound library to handle the playback of musical notes when a key is pressed.
3) Visual feedback on the piano keys when they are pressed. Display the current state (e.g., recording, playback) to the user.

# Trying to Solve it by Yourself

Suggest various possible data structures as a solution to the problem

You will not use the Data Structures just like they have been taught to you.
Modify them!

Discussion, cross validation, use pen and paper

# Trying to Solve it by Yourself

Rigorous analysis of the proposed solutions

Trying it out the algorithms on different inputs, handling ALL the cases

Drawing pointer diagrams IS A MUST

# Trying to Solve it by Yourself

Flow of code, how will the function calls occur

Design all the necessary function prototypes,

Which parameters will they take as input
What will they return

# Prerequisites for Coding

If possible, learn how to use github

How to write well commented code

Implementing basic forms of the data structures to be used

Pointer diagram should be EXTREMELY clear

# Prerequisites for Coding

Ensure that your project guide has approved the final data structure

Consult atleast two professors before going before proceeding to the next step

# Code (Easiest part)

Code should be generic, reusable and readable

First write the pseudo code on paper !

Assign proper names to the variables, functions, data structures

Keep separate .h and .c files

# Code (Easiest part)

Fix the errors yourself, don’t take help from intelligences, websites

Write good comments

Keep the code well formatted

Follow one style of writing code, keep it consistent

# Analysis of Code

Try to run various input data sets against your code and take account on parameters like:

- Accuracy
- Time Taken
- Many more..

Try plotting these obtained values for better analysis. This will help to improve your algo further.

# Reference

Feel free to reach out all faculties, seniors and us too :)

Arnav Prasad
9359026619

Arjun Deodhar
9730958430