<https://www.tellabs.com/>

<https://www.nixu.com/>

<https://github.com/TT00FE39-3001/lecture2>

55d41251

<https://www.youtube.com/watch?v=aGjL7YXI31Q>

# Algorithms Lecture 1 -- Introduction to asymptotic notations

<https://www.youtube.com/watch?v=zWg7U0OEAoE>

# Lecture - 1 Introduction to Data Structures and Algorithms

# Topics

* Review
* Algorithm techniques
  + Brute Force
  + Decrease-and-Conquer
  + Divide-and-Conquer
* Analyses tool(s): Big �

## Part 0: Review

* Course mechanics
  + Activities: Group + Individual (at home),
  + Due before next lecture
  + Approach: We try to present the material in an object-based approach (not OOP), therefore we will not use inheritance etc.s
* Previous lecture:
  + [Abstract Data Types(ADT)](https://en.wikipedia.org/wiki/Abstract_data_type)
  + [Data Structures](https://en.wikipedia.org/wiki/Data_structure)
  + **Array-based** [Stacks](https://www.softwaretestinghelp.com/stack-in-cpp/)
  + **Array-based** [Queues](https://www.softwaretestinghelp.com/queue-in-cpp/)

## Part 1: Brute Force, Decrease-and-Conquer

* Algorithm techniques:
  + [Brute Force](https://www.geeksforgeeks.org/brute-force-approach-and-its-pros-and-cons/): [Linear Search](https://www.softwaretestinghelp.com/searching-algorithms-in-cpp/) of lists
  + [Decrease-and-Conquer](https://www.geeksforgeeks.org/decrease-and-conquer/):
    - [Binary Search](https://www.softwaretestinghelp.com/searching-algorithms-in-cpp/) of sorted lists
    - [Running time of binary search](https://www.khanacademy.org/computing/computer-science/algorithms/binary-search/a/running-time-of-binary-search)
* [Recursion vs Iteration](https://www.geeksforgeeks.org/introduction-to-recursion-data-structure-and-algorithm-tutorials/)
* [Activity 1](https://github.com/TT00FE39-3001/lecture2/blob/main/activity1/README.md)

## Part 2: Analyses tools

* Math review: [Logarithms](https://www.mathsisfun.com/algebra/logarithms.html)
* [Big $O$](https://justin.abrah.ms/computer-science/big-o-notation-explained.html)
* Tools: [Graphing online](https://www.mathway.com/graph)
* [Activity 2](https://github.com/TT00FE39-3001/lecture2/blob/main/activity2/README.md)

## Part 3: Algorithm techniques

* [Divide-and-Conquer](https://en.wikipedia.org/wiki/Divide-and-conquer_algorithm)
* [Activity 3](https://github.com/TT00FE39-3001/lecture2/blob/main/activity3/README.md)

## Misc

* [GitHub workflow](https://github.com/TT00FE39-3001/lecture2/blob/main/github.md)
* [Links](https://github.com/TT00FE39-3001/lecture2/blob/main/links.md)