# longest increasing subsequence

* In this topic, we will learn how to find the longest increasing subsequence in an array using dynamic programming. The longest increasing subsequence is a problem that is used to find the length of the longest subsequence from the given subsequences in which all the elements are sorted in increasing order.
* The Longest Increasing Subsequence (LIS) problem is to find the length of the longest subsequence of a given sequence such that all elements of the subsequence are sorted in increasing order.

**Example**

* In the first 16 terms of the binary Van der Corput sequence
* 0, 8, 4, 12, 2, 10, 6, 14, 1, 9, 5, 13, 3, 11, 7, 15
* a longest increasing subsequence is
* 0, 2, 6, 9, 11, 15.
* This subsequence has length six; the input sequence has no seven-member increasing subsequences. The longest increasing subsequence in this example is not the only solution: for instance,

0, 4, 6, 9, 11, 15

0, 2, 6, 9, 13, 15

0, 4, 6, 9, 13, 15

**Data Structures**

A data structure is a particular way of organizing data in a computer so that it can be used effectively.

**Stack Data Structure**

# \* Stack is a linear data structure that follows a particular order in which the operations are performed. The order may be LIFO(Last In First Out) or FILO(First In Last Out).

Mainly the following three basic operations are performed in the stack:

push: Adds an item in the stack. If the stack is full, then it is said to be an Overflow condition.

Pop: Removes an item from the stack. The items are popped in the reversed order in which they are pushed. If the stack is empty, then it is said to be an Underflow condition.

Peek or Top: Returns the top element of the stack.

isEmpty: Returns true if the stack is empty, else false.

\* **Applications of Queue Data Structure**

Queue is used when things don’t have to be processed immediately, but have to be processed in First In First Out order like Breadth First Search. This property of Queue makes it also useful in following kind of scenarios.

1) When a resource is shared among multiple consumers. Examples include CPU scheduling, Disk Scheduling.

2) When data is transferred asynchronously (data not necessarily received at same rate as sent) between two processes. Examples include IO Buffers, pipes, file IO, etc.

3) In Operating systems:

a) Semaphores

b) FCFS ( first come first serve) scheduling, example: FIFO queue

c) Spooling in printers

d) Buffer for devices like keyboard

4) In Networks:

a) Queues in routers/ switches

b) Mail Queues