Stack can be implemented using an array or a linked list. Stack is a collection of elements with certain operations Flowing LIFO (last in first out) discipline. If our stack is full Then it means we ca n't add more elements to it. If stack is empty It means we remove elements from it. It has no element inside. So these two functions determine that That is our stack empty or full ok. Those who have read it, tell to others also. And if you share the playlist Then I will feel good. Sadly very less people in fact two people have shared playlist. Ayush did it and someone named Gautam did it. So thank you from the bottom of my heart. If you share then it will reach to many students.

Implementation Stack using Array

Moving forward with our data structure algorithm discussion we will see how to implement stack with the help of array. If you have not seen my last video then in that video I have told you what stack data structure is. and why we would want to use this type of data structure. I will come back here and in our discussion I will show you stack. Top is the value of index. It is not a pointer. Top 's default value will be -1 if there are no elements. If there are two elements in this then top will be 1. Means my stack is filled up till here. And I will delete this and I will use this array only , my array is holding two elements so my top is filled in here in stack. The top value will be 0 if my stack is filled till here, 1 if till here and 2 if there is no element in this and with that I 'll write s. size is equal to 80. The default value is -1 but it will store the index of topmost element. This way I have implemented my stack. Can create array then can set top and then my stack will be ready.

We use integer as data type because it is the simplest way to understand things. But we can create custom data types stack too. It can be that whatever you are putting inside stack can be classes of C++. So I want to ask you one thing that how will you pop and push an item ? Then what you will do ? So which method will be best for me ? Most of the operations are performed in O (1) Operation can be complicated too. So I will write most of the. operations here. In O (1) means most of. operations are done in constant time. Now suppose I want to add 8 in it so I write it here and I will add 1 in this so this will become 1. And I did my push operation so we chose this and not this. So for pop operation then give me II if I show you pop operator what it does. To implement stack with the help of array what I did first ? First of all I created a fixed size array you can see here. In this stack , I created array , size and top which tells us the size of array. Then I created stack and kept its size 80 so that if I need to do a big number of push operations then my stack can handle it without problem so this is how we implemented stack here. I hope this is clear to you and this is my structure which I have created.

See in code. So this stack we have rotated it in this direction and make it standing. And we have choose this orientation because if I want to push any element in this Then I would have difficulty to maintain all the indexes and that would not be ideal. So in this way I chose here 0 , 1 and my stack is growing in the direction. I hope this is clear and we will write its code and then we will understand it 's operation in more depth.

C code for Implementation Stack