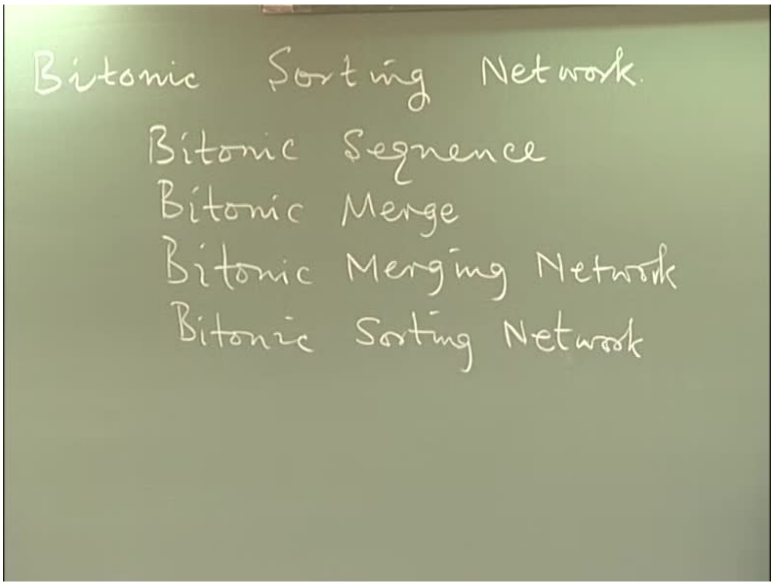
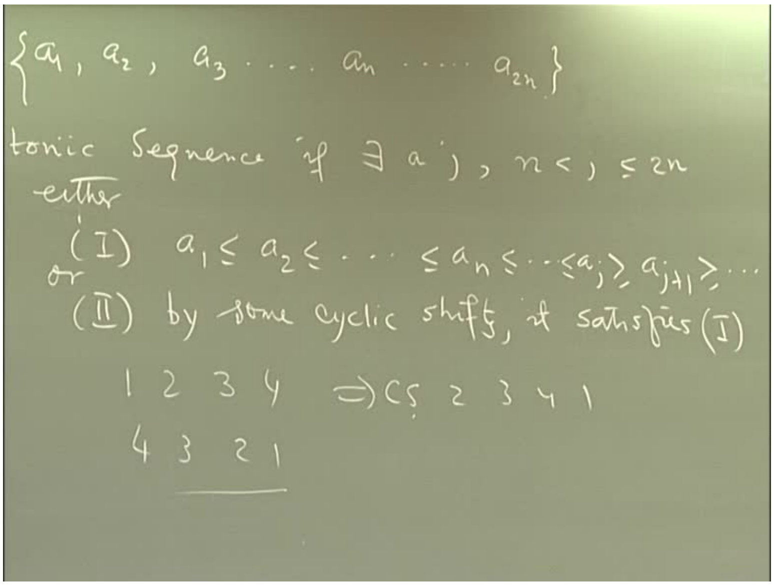
**Parallel Algorithm**
**Prof. Phalguni Gupta**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Kanpur**

**Lecture - 8**

(Refer Slide Time: 00:22)



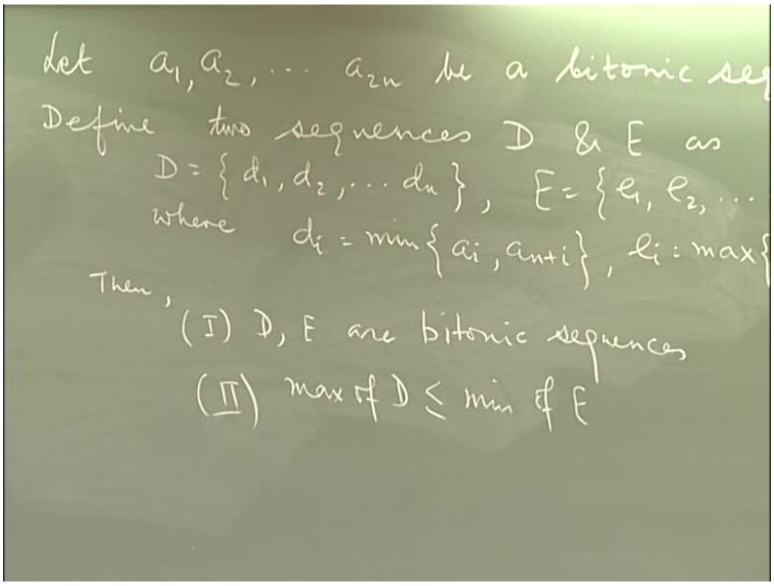

(Refer Slide Time: 01:42)



For today, we will be discussing on the Bitonic sorting network. Now, same as did in the earlier case, here also we will first define what is bitonic sequence and then, we will be discussing bitonic merge. Then, we will be discussing bitonic merging network and finally, will have bitonic sorting network. So let a 1, a 2 be a sequence; a 1, a 2 to a n be a sequence. Now, the sequence is bitonic sequence, if there exists a j, such that, either a 1 less than equal to a 2 n or by some cyclic shift, it satisfies 1.

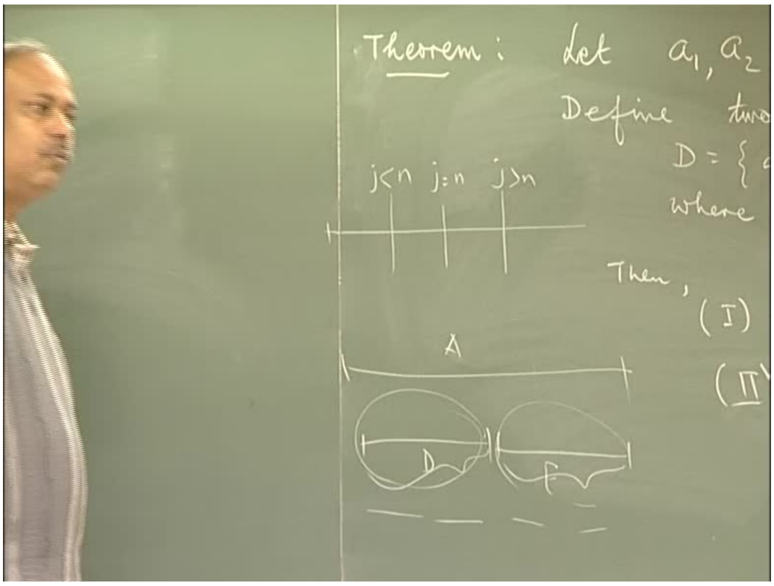
The second sequence, bitonic sequence, if either a 1 less than equals to a 2 less than equals to a n less than equals to a j and a j greater than equal to a j plus 1 is greater than equals to a to n or by some cyclic shifts, which is this one. Now, for example, a sorted sequence is a bitonic sequence, say why it is so? 1 2 3 4, so by cyclic shift, you can get 2 3 4 1 and which satisfies your (( )).

(Refer Slide Time: 05:23)



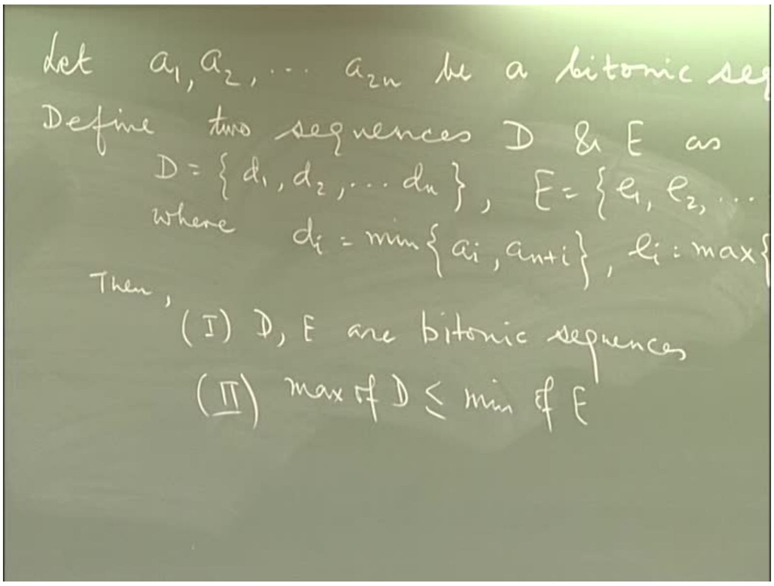

Similarly that 4 3 2 1 is also a bitonic sequence. This operator, even though I have written less than equals to, it can be greater than equals also. For a cyclic shift, it can show that one. Another thing what I can tell is, a sequence of 2 elements is also a bitonic sequence. A sequence of two elements is a bitonic sequence. So, definition is k s (( )) stage? No. Yes or no? If it is clear, then you can define a theorem. Let a 1, a 2 n be a bitonic sequence of 2 n elements. Define two sequences b and e as, where d i is minimum of a i and a n plus i and e i is maximum of e i and plus i. Then, d and e are bitonic sequences. The maximum of d is less than or equals to minimum of e. So statements are clear?

(Refer Slide Time: 08:35)



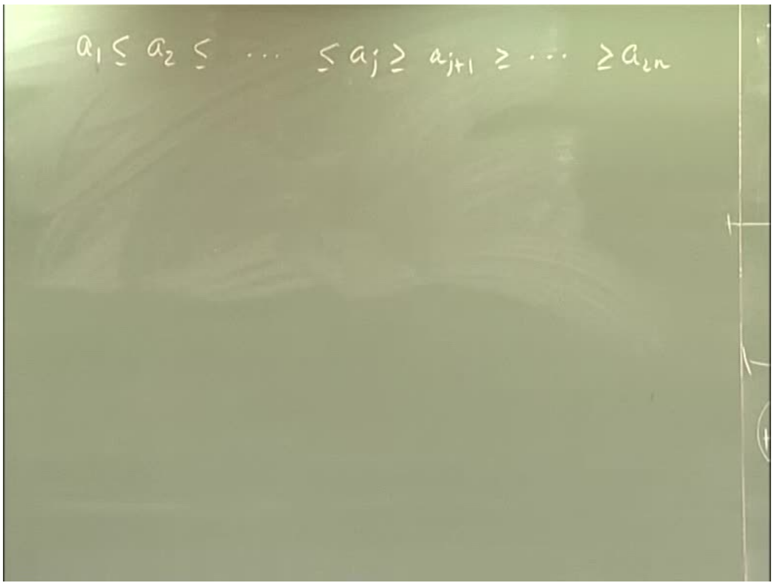

What we have obtained? We have obtained two sequences d and e. d is obtained and the i th element d is obtained by taking the minimum of these two elements and e i is obtained by taking the maximum of these two elements. Then, d is a bitonic sequence and e is also bitonic sequence and maximum of d is less than equals to minimum of e. Now, in what, why this can be used for sorting? See, what it gives you that, from this sequence, you are making two sequences, one is d and another one is e, right. From these two sequences, you make d and e. Now this, recursively you do the bitonic merge on that, recursively you do the bitonic merge on this and you get another subsequent like that. Every time you are achieving one thing, then these elements are smaller than these elements. These elements are the smaller than these elements. So, by some recursive method, if you can sort this one, then the elements become a sorted one. Is that ok? So, that is the idea. Now, how to prove this theorem?

(Refer Slide Time: 05:23)

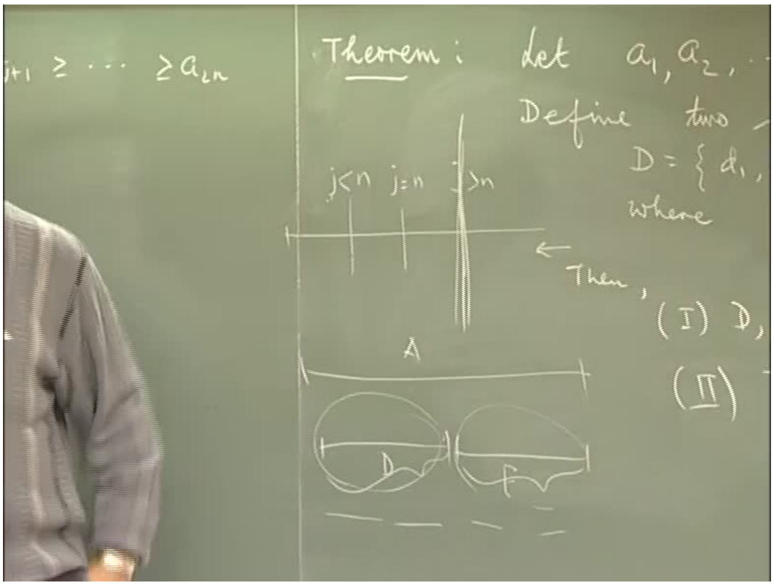



You have j equals to n and j less than n and j greater than n. So, first thing is that, the cyclic shifts of the sequence a will affect both the d and e sequence similarly. Though that I think the property of, these two property is, if you do the cyclic shifts, it will affect both the sequence d and e, right. Agreed? But it will not affect these properties, right.

(Refer Slide Time: 10:54)

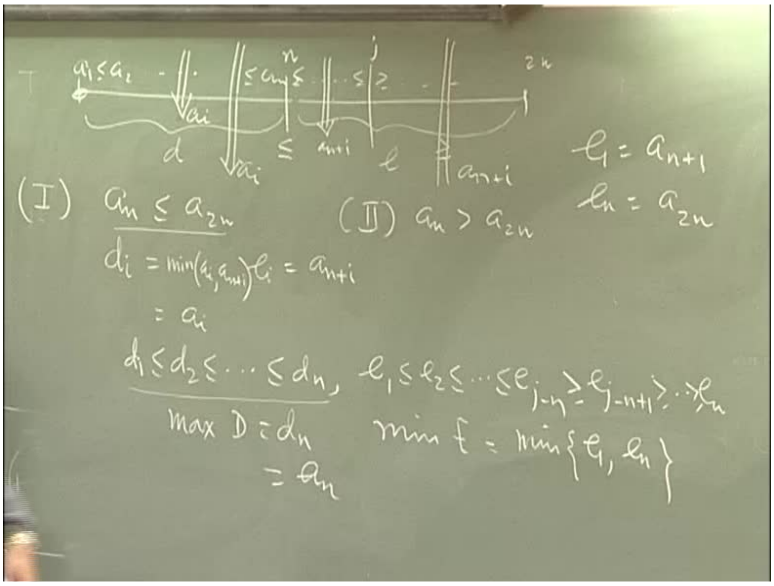


(Refer Slide Time: 11:21)




So, we assume that for simplicity, we assume the sequence is satisfying this type of characteristics. First assumption we are making. Now, second part is that, as he mentioned, the j can be less than equals to n and j can be greater than n, right. Now, both the cases, if it is less than n, I can see from this direction, that this j becomes greater than n. See, if I look from this side, agreed? Understood? If j less than n, if I see from that side, j becomes greater than n and that also becomes bitonic sequence. Only the less than symbol becomes the greater than symbol. Nothing else, right.

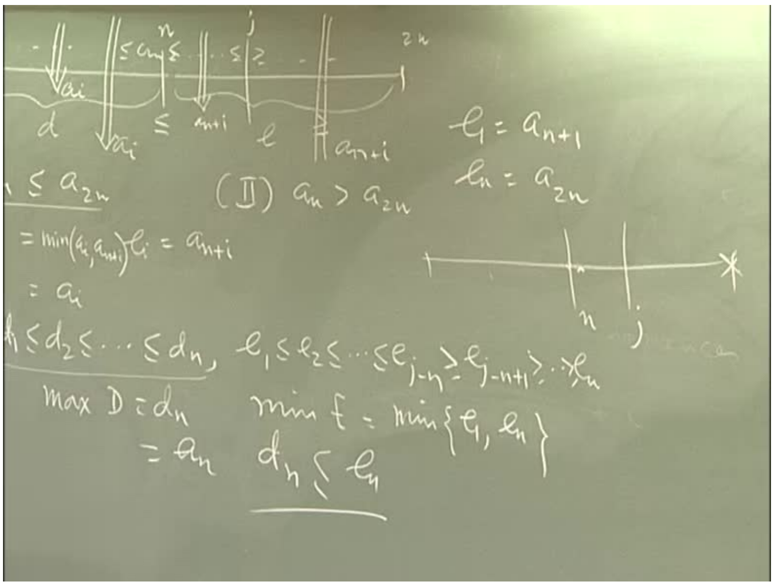
(Refer Slide Time: 12:18)



So, without any loss of generating, we assume that j is greater than n. If I can show this one, the other will also you can show. So, you assume that j is greater than n. So, you have a 1 less than equals to a 2 less than equals to a n and less than equals to a j plus 1, a 2 n. Now, up to this we have done it. This side is less than equal and greater than equal, right.

Now, assume that first case is, the two possibility cases, a n less than a 2 n and another case could be a n greater than a 2 n. These are the two possibilities you have. This is a n less than equals to a 2 n or a n greater than equal to a 2 n. Now, let us assume that a n greater than equal to a 2 n; first case. In that case, what is your d i and a i. d i is what? Minimum of a i and a plus i. So, what is that a i and a plus i? Who is the minimum? There is no ambiguity. Yes. So, I can write a i. What is a i? a n plus i. Agreed? Basically, this becomes your d sequence this becomes your e sequence. Yes. So, this satisfies d 1 less than equals to d 2 less than equal to d n and you want, less than equals to e 2 less than equals to; yes, because this side is (( )) less than equals to a j minus n and then, it is greater than. So, this is a bitonic sequence. So, this is a bitonic sequence and this also is a bitonic sequence. For this case? This case? Here, a 1 less than equals to a 2 less than equals to a n less than equals this greater than equals to and so on, right. Now, this element, this is your a i and corresponding element is, may be here.

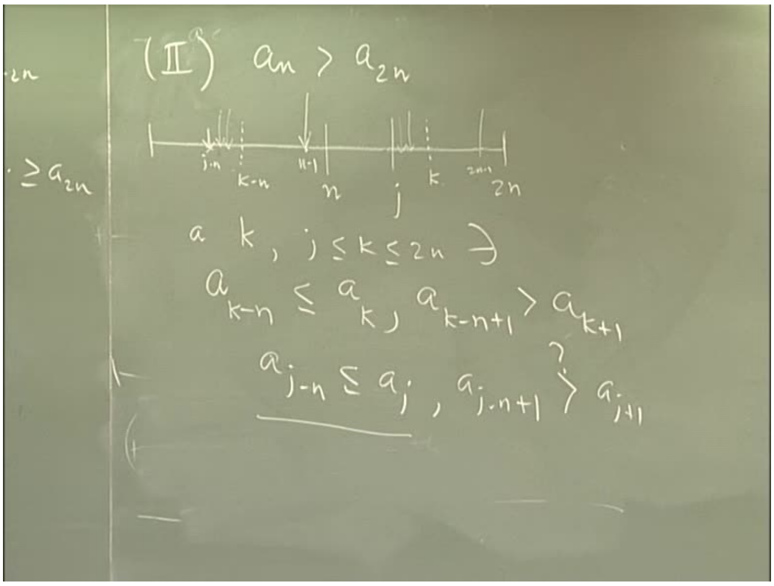
(Refer Slide Time: 18:22)



Suppose corresponding element is here, a n plus i, right. If it is the case, so this is less than less than less than, so this is larger than this one, so minimum becomes d i and maximum becomes here. Another possibility is that, your a i is here, a i plus 1, i is here, a and plus i is here. This element is less than this element. This element is less than this element. This element is less than this element. So, this element is less than this element. So, this becomes d i and this become c i. Agreed? So, this becomes d sequence and this becomes e sequence and both of them are bitonic sequences.

Second part is, the maximum of this sequence is d n and minimum of e sequence is what? Yes, minimum of e 1 or e n, right. Now, a 1 is either e 1. E 1 is what? e 1 is a n plus 1 and e n is a 2 n. Agreed? Now, if it is the case, the d n is equal your a n. You already have shown or from the definition that a n is less than equal to n plus 1. From the definition itself because, j is somewhere and this is your n and this your j. This element is always greater than this element and also by this case; this element is greater than this element. Agreed? So, d n is always less than equal to e n.

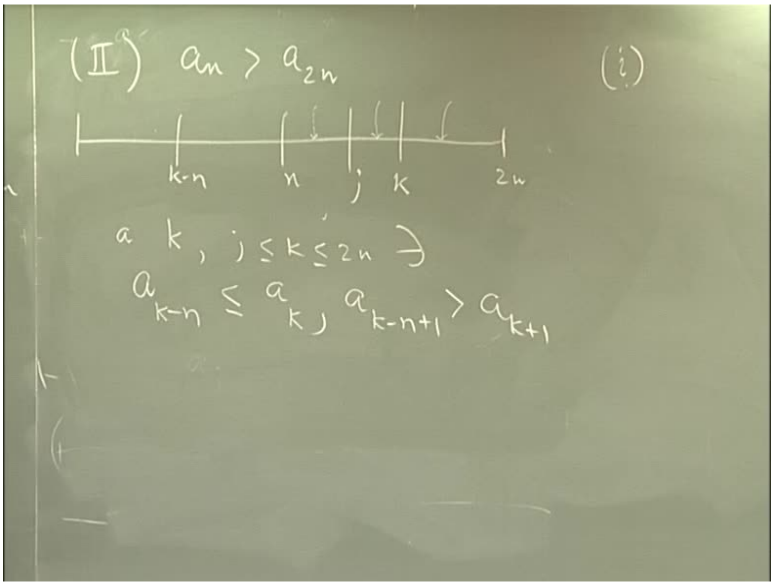
(Refer Slide Time: 18:49)



The next case is, is I can show for this case also, and then your problem is solved. This is given to you. Now, I can always find a k. In such a k, such that, a, I can always find out the k, such that, a k minus n less than equals to a k and a k minus n plus 1 greater than equal to a k plus 1. Now, how can I justify this? What I have found that a j minus n is less than equals to a j. This is given to you. Agreed? a j minus 1 is less than equals to a j. Because, from the definition of your bitonic sequence.

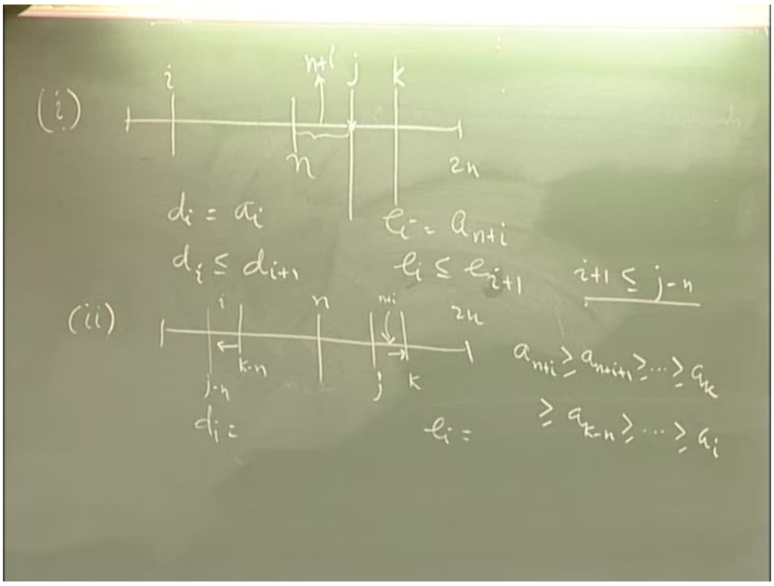
Now, you look for a j minus n plus 1, whether it is greater than a j plus 1 or not. Suppose this is your j minus n and this your j. So, this satisfies the property. This is less than equals to this. Now, whether this element is larger than this element and supposes it is not. That means, this element is less than equal to this element. You look for the next one, whether this element is larger than this element or not. Not. Suppose no. But, in that case, think about this one, n minus 1 and 2 n minus 1. So, this element is less than equals to this element, right. Because, you are going one by one, so you got the last case a n minus 1 is less than equals to 2 n minus 1.

(Refer Slide Time: 23:00)



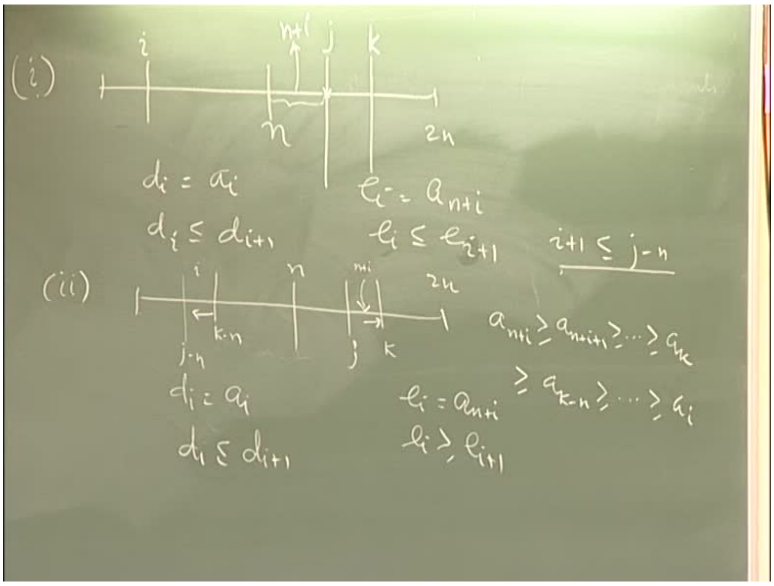

But, from the assumption, a n is greater than 2 n. So, you will get one k satisfying this criteria. I am looking for the first (( )). We are looking for the smallest case satisfying this criteria. Clear?

(Refer Slide Time: 23:50)



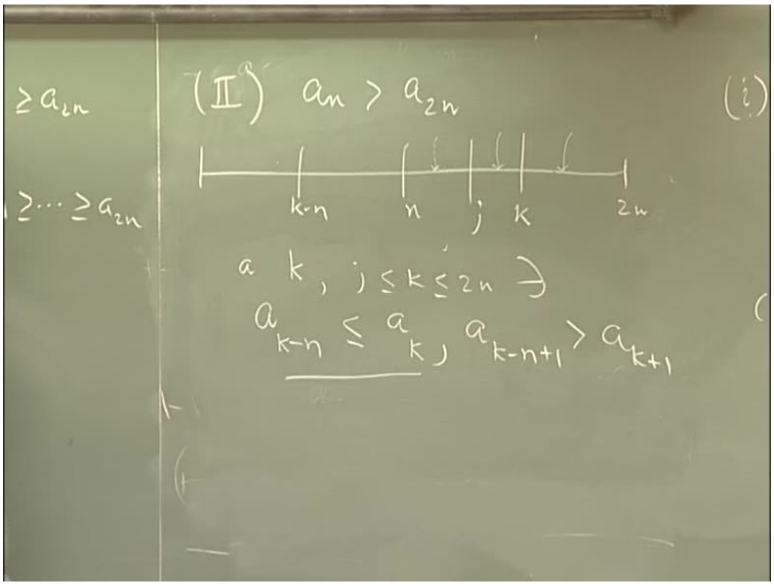

If it is clear, then you got, this is n, this is j, this is k, this is k minus n and this is 2 n. Now, here you the three cases. About your a i a n k and plus i. i can be, n plus i can be here, can be here and can be here. I can be here and I can be here. So, I can be here, here or here. Agreed? Now, suppose i is here. First case is, i is here. So, what should be the; this is your n, this is your j, this is your k and i is here, n plus i and i is here. This is the first case. Now, you tell me what should be d i and what should be your e i in this case? When i is n plus i is here. d i is a i and e i is a n plus i. Also, in this zone, while I am, n plus i is in this zone, can I write d i is less than equals to d 1 plus 1?

(Refer Slide Time: 27:21)



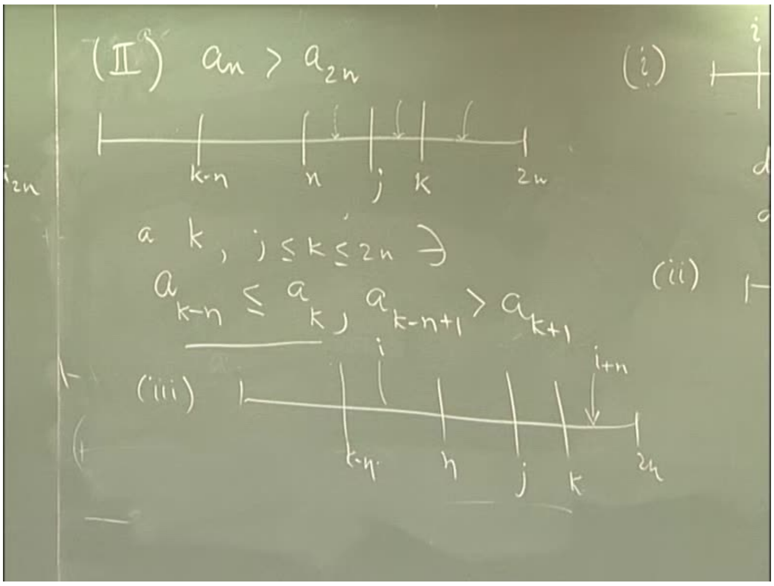
e i will also be e i plus 1. I can right this, i less then equals to this number. Not crossing this number. I can write i plus 1 less than equals to j minus n. That condition is proved. So, that, up to this point, this point I am talking about. Is it ok? Yes. Now, let us take the second case, 2 nm this is your j, this is your k and this is your k minus n and n plus i is lying here. Now, can you tell me, when n plus i is lying, what is your d i and what is your e i? So, if you shown that, if you are writing like that, a n plus i is greater than equal to, a n plus i plus 1 greater than equal to a k. a k is greater than equals to a k minus n greater than greater than equal to a i.

(Refer Slide Time: 27:31)



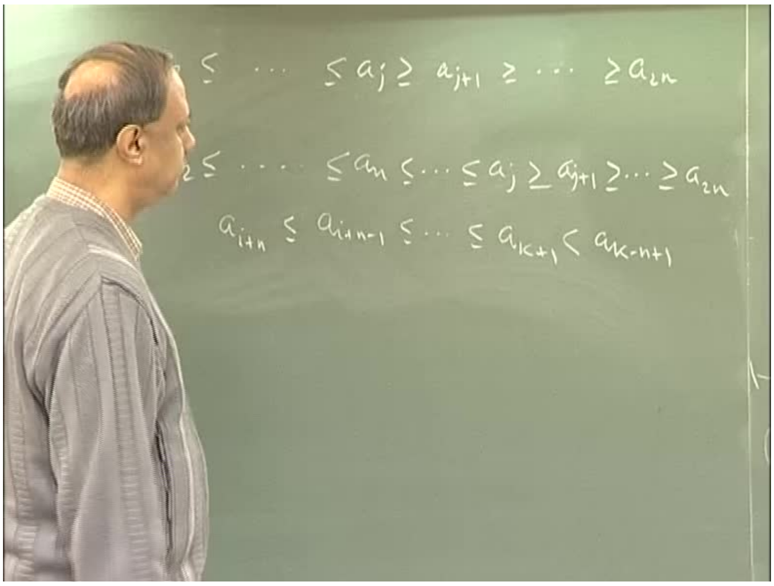
Basically, I am going from this direction to this and then, from this to this. I have used this property, right. So, you are writing d i is your a i and e i is a i plus i and what is the relationship between d i and d i plus 1 here? d i is and what is d i? So, you got the second thing.

(Refer Slide Time: 28:06)



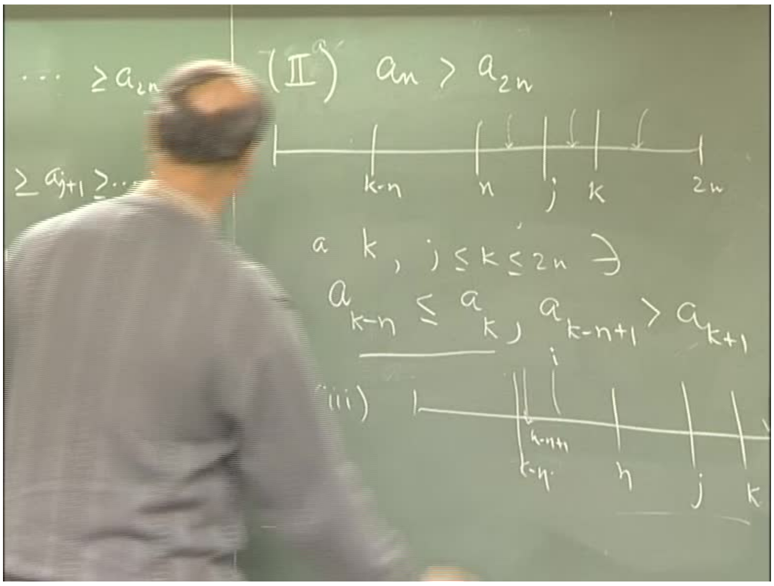
Now, I need the third case. Third case is, here j, here k, k minus n and your i is lying here and i is here.

(Refer Slide Time: 28:37)

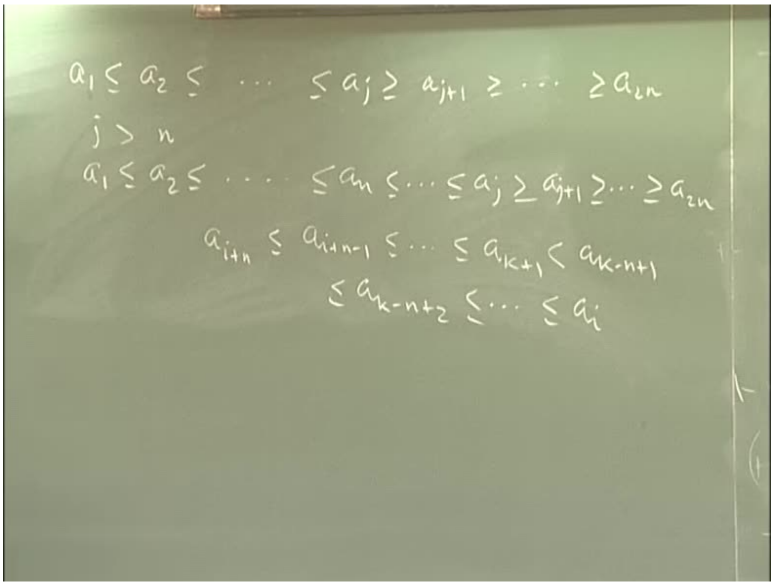


Now, you see, here I can write a i plus n is less than equals to a i plus n minus 1 a k plus 1 less than a k minus n plus 1.

(Refer Slide Time: 29:05)

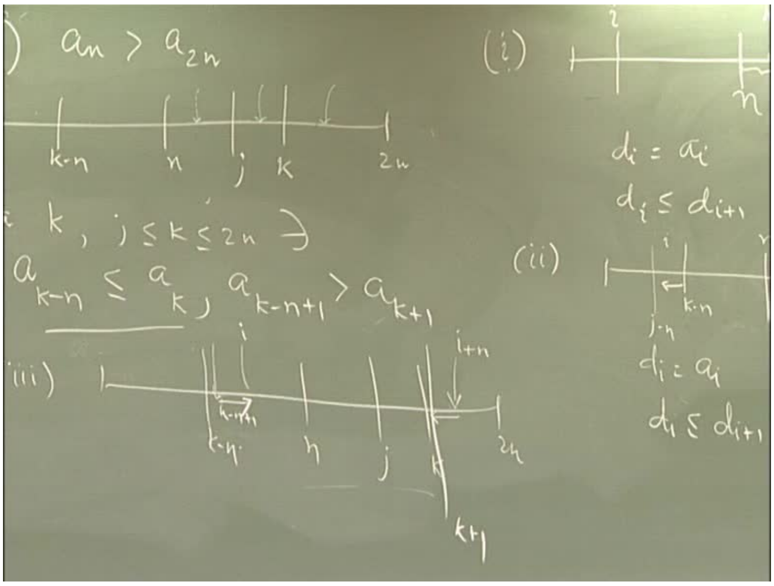


(Refer Slide Time: 29:11)



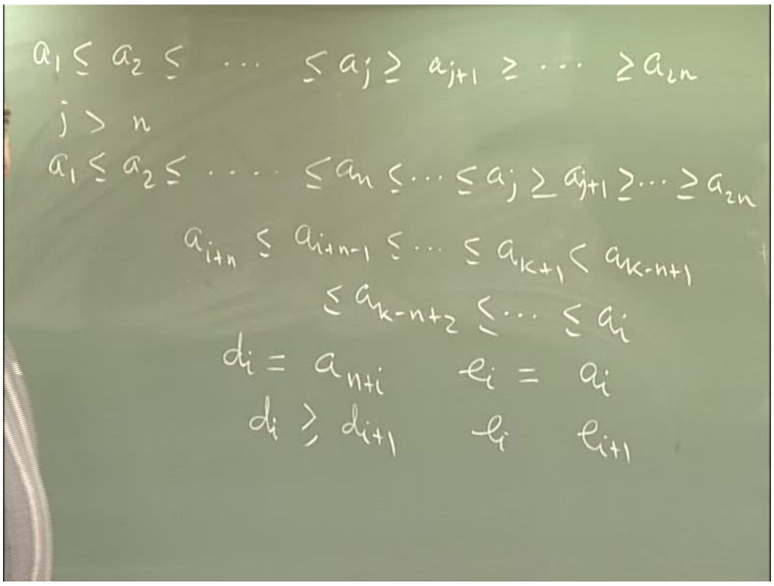

So, it is here and less than equals to a k minus n plus 2 a i. Agreed?

(Refer Slide Time: 29:23)



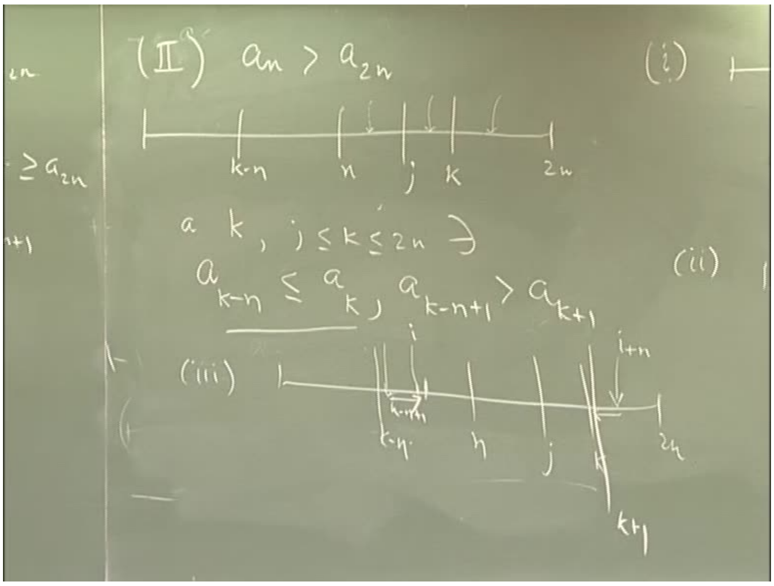

Basically what I am doing, I am moving from this side to this side. Here, it is k plus 1 and then, moving from this to this. From this side to this k plus 1 and I know the relationship between k plus 1 k minus 1 and plus 1.

(Refer Slide Time: 29:49)

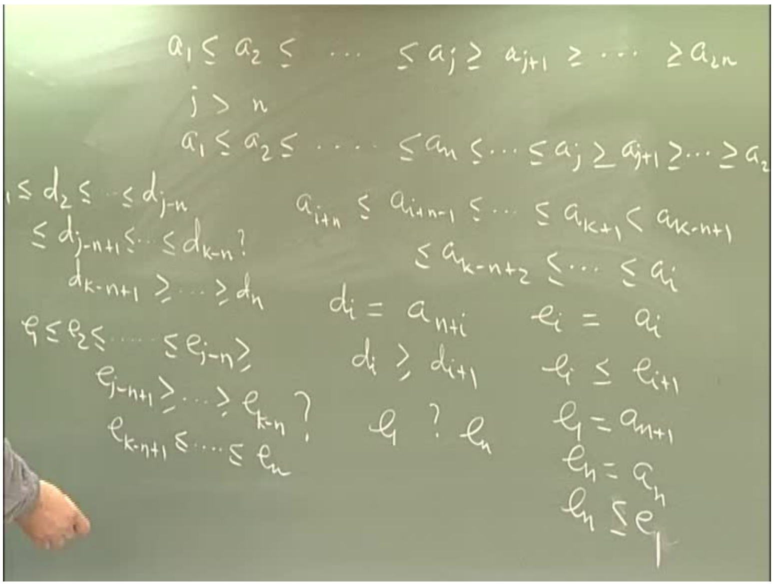



I am moving from this to this. Is it ok? If it is ok, then here I get this relation. From here, I get d i and e i. What you get? a n plus i and here a i, right. What is the relationship between d i and d i plus 1? e i and e i plus 1? d i is greater than equals to, yes, because a i plus 1 will be here.

(Refer Slide Time: 31:07)



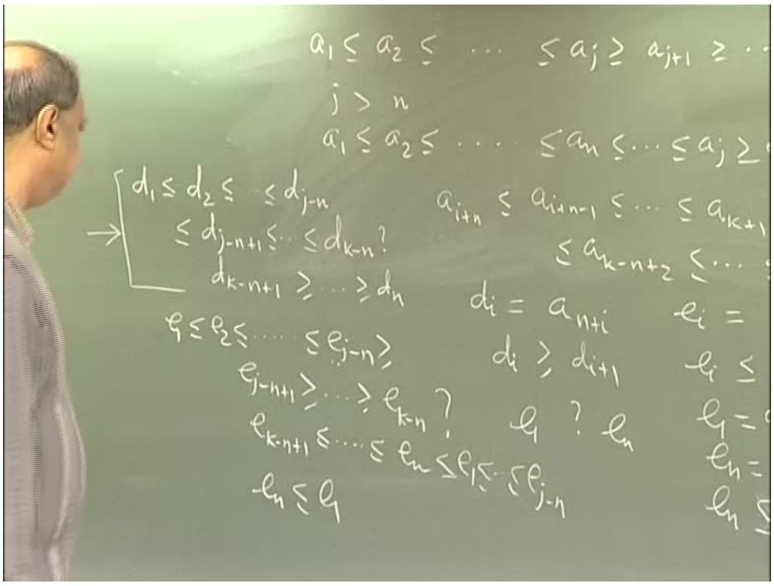

(Refer Slide Time: 31:53)




So, this is greater than this and e i? e i? a i? Why? See, a i is coming this side, right. a i is this one and a i plus 1 is this. So, a i is this. Is it not? This is a i and this is a i plus 1. So, if I combine all this, what I get? First, let us consider the d 1 case. You get d 1 less than equals to d 2 less than equals to d j minus n less than equals to d j minus n plus 1 d k minus n. Then, I do not know the relationship here, d k minus n plus 1 d n, right. I do not know the relationship here, because here it takes up to d k minus n and this side, and I am starting from d k plus n minus 1. I do not know that relation or what would be the relation. So, the relation we will find out what is the relation. Up to this is known to you.

What about e 1? 3 1 is less than equals to e 2 less than or equal to e j minus n. Then, you have greater than equals to e j minus n plus 1 greater than equal to up to e k minus n. I do not know the relationship between e k minus n and e k minus n plus 1. Then, I know this relation.

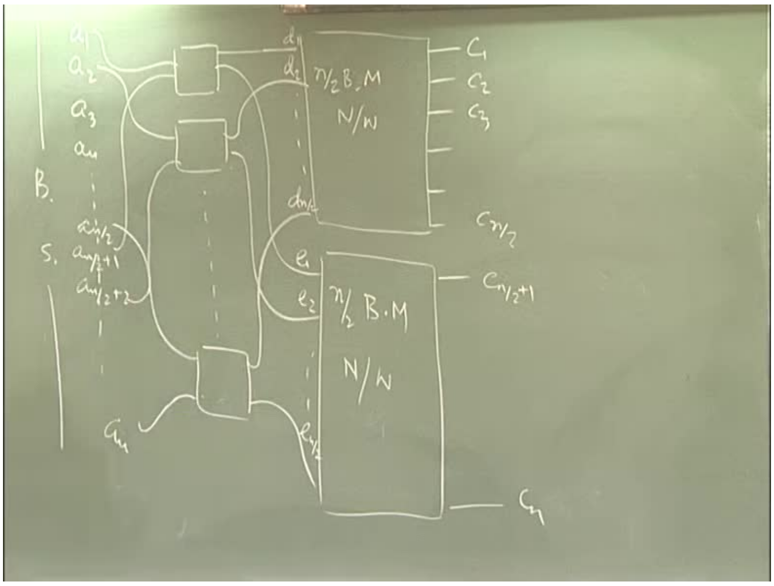
(Refer Slide Time: 35:43)




Do you know any relationship between e 1 and e n? What is the relationship between e 1 and e n? What is your e 1? e 1 is; what is e 1? a n plus 1. e 1 is a n plus 1. What is e n? e n is where? If it is, do you know any relationship between e 1 and e n? So, I can write e n is less, sorry e n is less than e 1. Agreed? Yes. Now, in that case, first you tell me, if I know how this is, now I know the relation here. e n less than equal to e 1. So, think about this one? Can I tell this is a bitonic sequence irrespective of whatever is the relationship between these two?

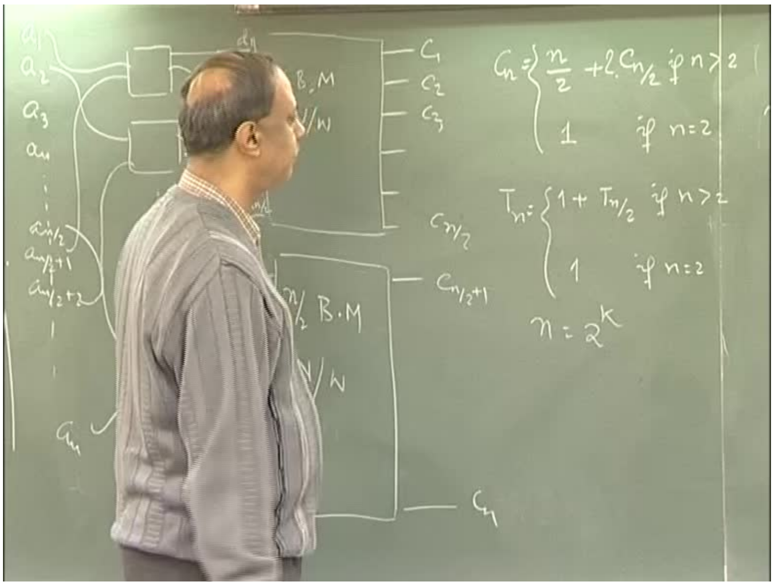
Suppose it is less than equals to, then it is a bitonic sequence. If it is greater than or equal to, then also it is a bitonic sequence. So, this is a bitonic sequence. What about this one? This is also bitonic sequence, because by cyclic shift, I can find out this is less than equals to e 1 less than equals to up to less than equals to e j minus n, right. Then, I can get e j minus n plus 1 greater than greater than greater than e k minus n. Whatever is the relationship between this, again less than equal to this, so it becomes a bitonic sequence. So, the first point is that, d and e they are bitonic sequence. That is true.

(Refer Slide Time: 36:55)



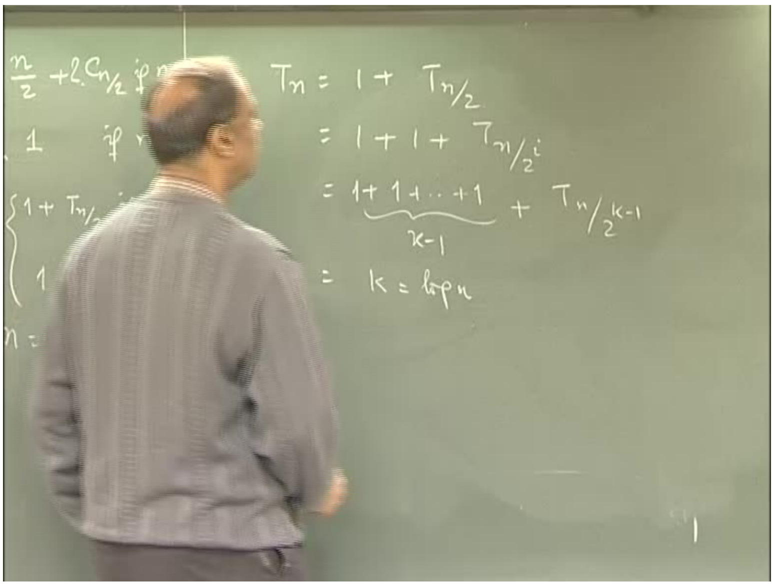

Now, what you have to show? That maximum of d is less than equals to minimum of d. Now, suppose you have a bitonic sequence, a 1 a 2 a 3 a 4 a n, right. What should be your network? Network will look like, that you have comparator n by 2 comparator and you have a n by 2 plus 1 n by 2 plus and 2 and so on. So, it goes like this. So, this is a bitonic system. This is a bitonic system. So, a i has to be compared with a n plus i. Same thing we did and then, you get n by 2 bitonic merging network and you have n by 2 bitonic merging network, right. So, you get here basically d 1 d 2 d n by 2 and you get e 1 e 2 e n by 2. Then, recursively if you call, you get c 1 c 2 c 3 c n by 2 plus 1 c n. So, you have one comparator and a i is compared with a n by 2 plus i and smaller elements you keep it in this sequence and larger elements, bring it to e sequence and then, this becomes a bitonic sequence. This become a bitonic sequence recursively you do it and then, finally you get sorted sequence in this side and sorted sequence in this side, right.

(Refer Slide Time: 39:39)

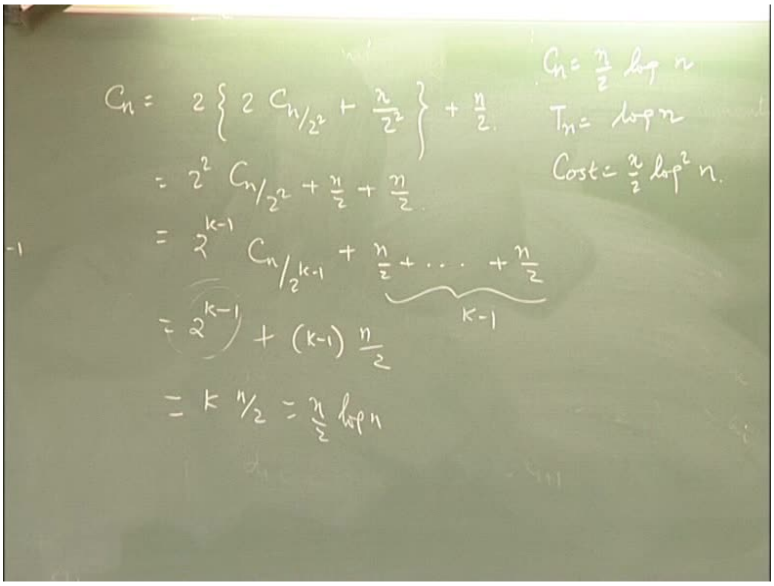



If it is the case, then how many comparators you need? Number of component is n by 2 plus c n by 2 2 times. If n is greater than 2 and one is n equal to 2. n by 2 plus 2 n by 2, if n is greater than 2 and one, if n is equal to 2. Because, you have two elements and you have one comparator. The number of steps, parallel steps you need, t n is equals to 1 plus t n by 2, if n is greater than 2 and one is n equal to 2, right. So, there is one and then, whatever time you need. Now, assume n is equal to 2 k.

(Refer Slide Time: 41:12)



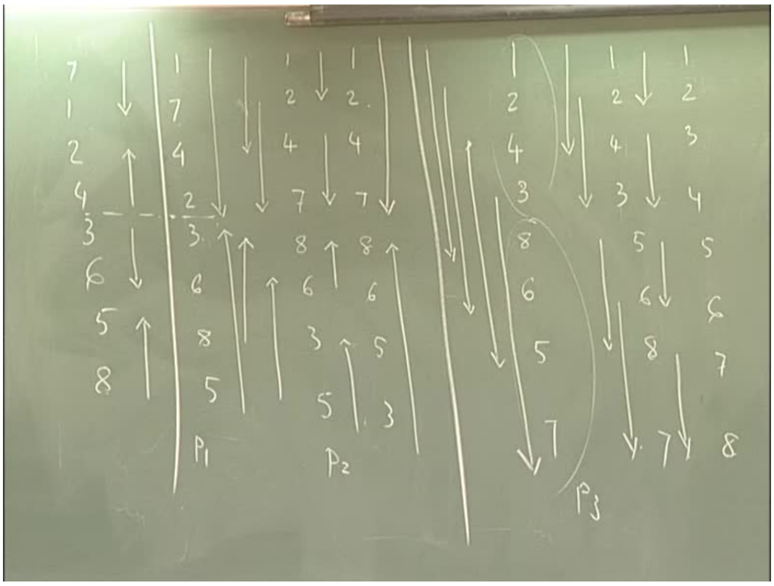
(Refer Slide Time: 41:53)



So, you can easily solved t n, because, this becomes t 2 and t 2 is also 1, so it is k, which is your log n. Now, if I have to compute, c n is equal to 2 times c n by 2 square plus n by 2, so you get 2 square c n by 2 square plus n by 2 plus n by 2. Then, 2 to the power k minus 1 c n by 2 k minus 1 plus n by 2. So, 2 to the power k minus 1, because c 2 is 1, so you get k minus 1 n by 2 and this is also 1 n by 2, right. So, you get k n by 2. That is, n by 2 log n.

So, you get c n is equals to n by 2 log n. T n equals to log n and cost is n by 2 log square n. So, you merge a bitonic sequence into a sorted sequence. One thing you the absorb that, you must have a bitonic sequence to merge this one. So, the sequence must be a bitonic sequence.

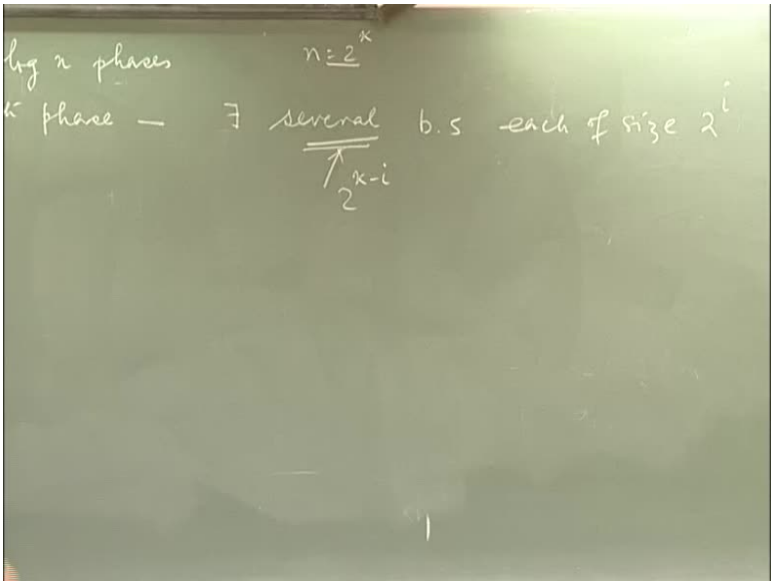
(Refer Slide Time: 44:07)



If I have to think that how can I use this idea to sort n arbitrary element. So, let us (( )) 8 elements, 7 1 2 4 3 6 5 8. Yes. Idea is that you use the alongs of the two elements are always bitonic and then, you merge it. You must get it, after merge, you get the larger bitonic sequence. So, basically it is nothing but, the idea is that, this one you may make increasing sequence and this you get in decreasing sequence, so that you get, at the end of this, you get a bitonic sequence of larger size. So, here you will be getting 1 7 4 2 3 6 8 5. See, you observe, now this is the bitonic sequence and this is another bitonic sequence. You can get this in sorted sequence in increasing order. This should be in increasing order and this should be decreasing order, so that this becomes another bitonic sequence.

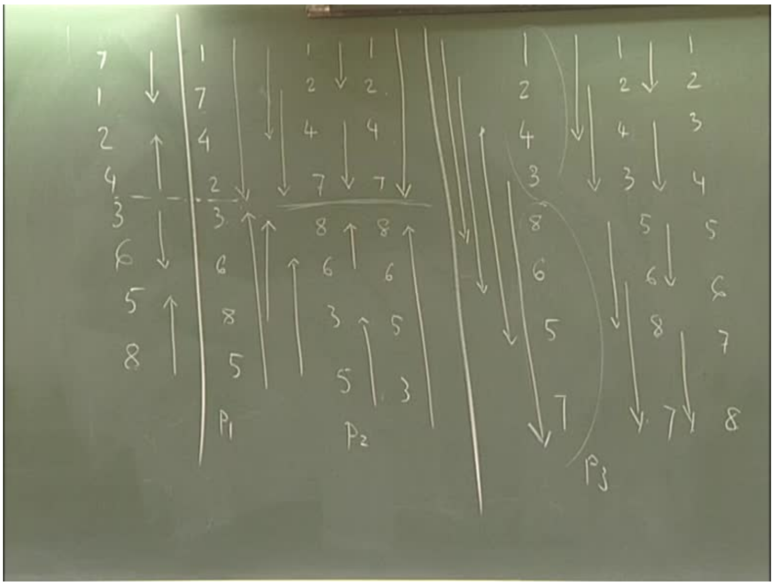
So, in order to do that, this should be in increasing order. You first compare, a i is to be compared with n plus i. So, you get here 14 2 7. Now, this becomes a bitonic sequence and this becomes a bitonic sequence, so you get a sorted sequence here. Similarly, is the case here. So, compare these two and you get 8, you get 3, you get 6, you get 5 and then, you compare these two. You get 8 6 5 3. So, this becomes another bitonic sequence. You observe, at the end of this stage, you get a bitonic sequence of size a. Now, you have to make it a sorted sequence. So, a i has to be compared. That is, this has to be compared, this has to be compared with this and this has to be compared with this and this has to be compared with this, right. After doing that, so if I compare 1 with 8, you will get 1 and here 8, 2 with 6, 2 here, 6 here and 4 with 5, 4 here and 5 here, 3 with 7, 3 here and 7 here. Now, the problem is, this is a bitonic sequence and this is another bitonic sequence and all these elements are larger than these elements, right. Because, d i is smaller than e i. So, you now compare this with this, this with this, this with this and this with this, you get here 1 4 2 3 5 8 6 7. Then, you compare with this, this 1 2 3 4 5 6 7 8. Now, how many parallel steps you need? Parallel phases you need? This is phase 1, phase 2, and phase 3.

(Refer Slide Time: 48:15)

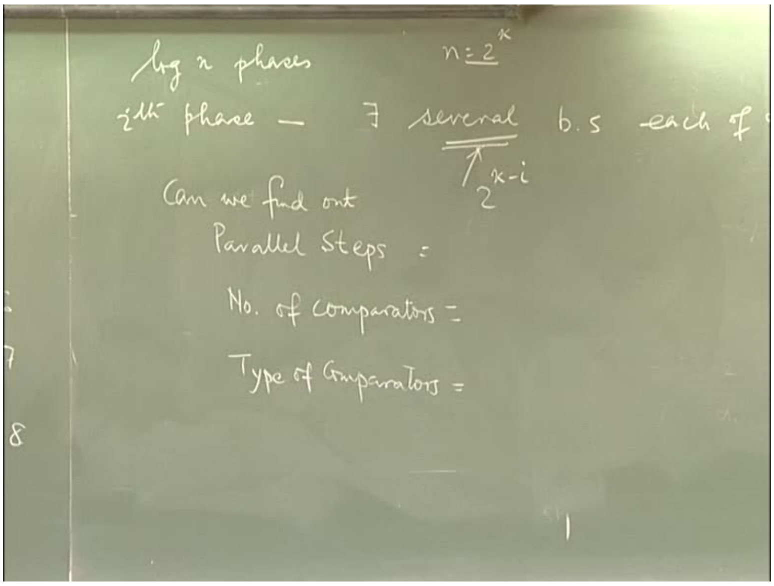



So, if you have a bitonic sequence, if you have the sequence of 8 elements, there will be 3 phases. That means, you have the log n phases. Agreed? At the i th phase, you have several parts, each of size 2 to the power i bitonic sequence. At the i th phase, there are log n phases. Agreed? At the i th phase, there are several bitonic sequence each of size 2 to the power i. Now, can you tell me what is this value of several? n by 2 to the power i.

(Refer Slide Time: 50:02)

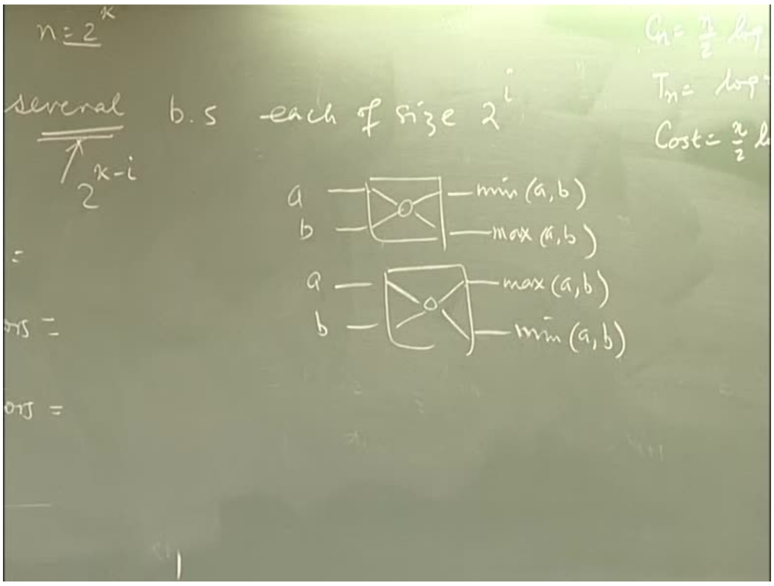


(Refer Slide Time: 50:28)



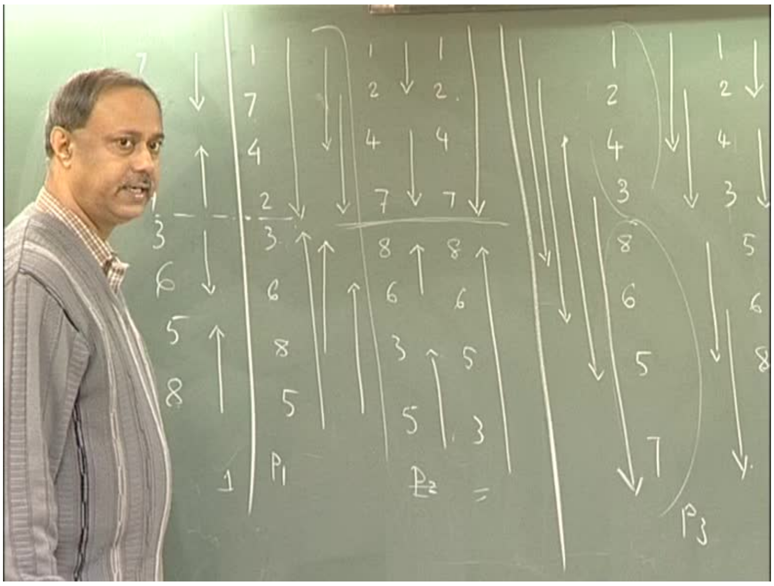

That is, 2 to the power k minus i. n is 2 to the power k, right. So, that many parts are there. In this case, second phase, you have 2 to the power i, 2 to the power 2 4 and there are two such cases. So, it is 2 to the power k minus I bitonic subsequences each of size 2 to the power i. Agreed? Now, if it is the case, can we find out total number of parallel steps and number of comparators and then, type of comparators.

(Refer Slide Time: 51:33)



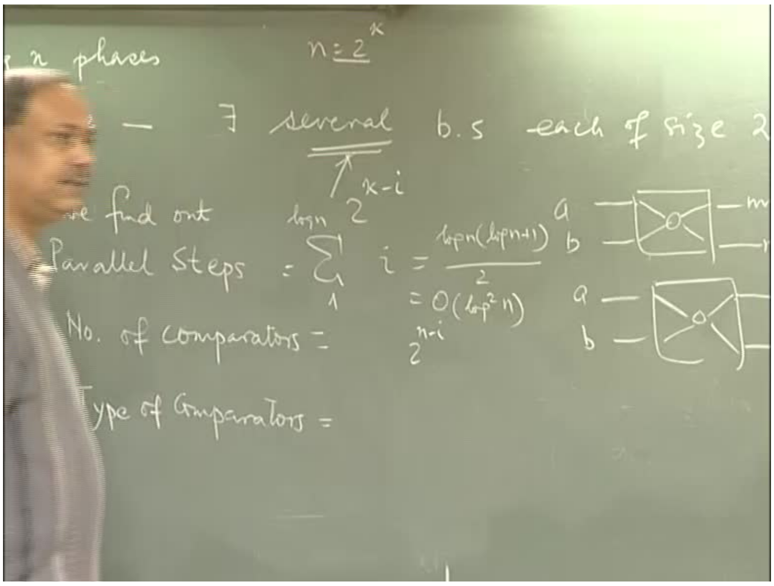

You need to know these three things. You observed that, here it is not like ordinary (( )) that the comparator has two inputs and the output is the, first output is the minimum and second output is maximum. Here, it is depending upon the phase, whether it is in increasing order or decreasing order, right. So, the two types of comparators you will be using a and b min. Actually, it is not very difficult to implement. The first one, if you implement only the (( )), you have to change to give you the maximum output in the top and minimum output in the final. This is not a big thing. Only thing is that, you have to, phase wise you have to keep it in mind which comparator you will be going to use, right.

(Refer Slide Time: 53:08)



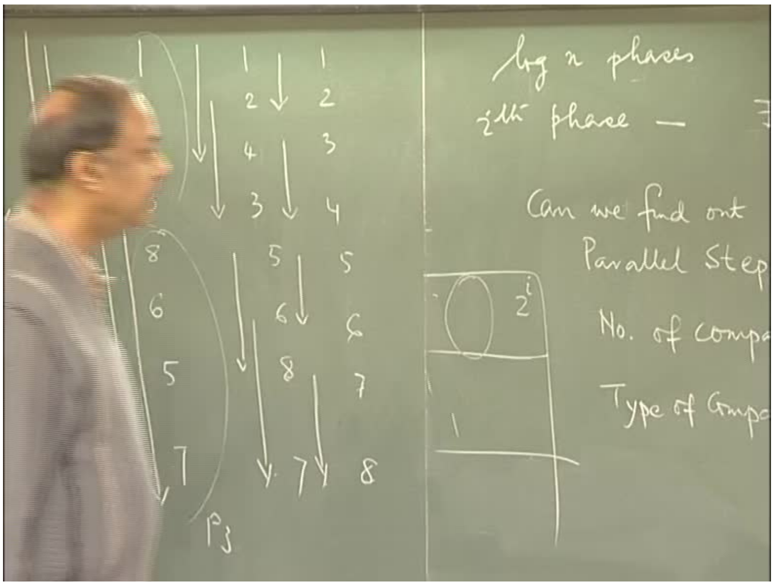
In the first phase, alternatively high low high low high low and here, in second phase, it is low low high high and then, this is low low low low. That is the only thing, right. So, this, easily you can do this. The type of comparators you will be using that is known to you. Now, what about these two? Number of parallel steps, first phase, one parallel step and second phase, how many parallel steps? right

(Refer Slide Time: 54:00)

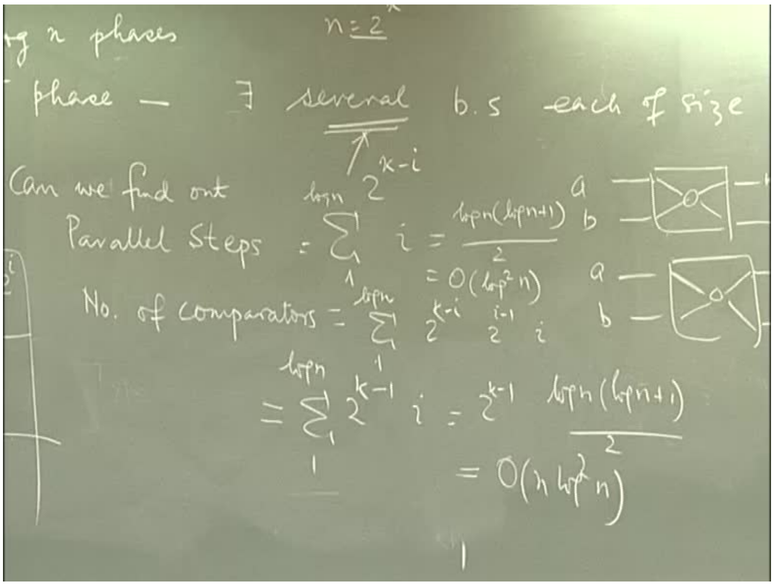



How many parallel steps? This is one, this is one and this is another one. Agreed? Third one is what? How many parallel steps? This is one. This number of parallel steps you have already obtained. The third step, you have 2 to the power 3 elements and they are bitonic sequence and you are making three parallel steps and the i th is you are making i parallel steps. Agreed? So, i parallel steps at the i th stage. How many i? i is 1 to log n. So, it is log n log n plus 1, which is order log square n. So, number of parallel steps is order log square n. Now, can you tell me how to find out number of comparators at the i th stage? At the i th stage, how many parts are there? 2 to the power i 2 to the n minus i, that many sub parts are there.

(Refer Slide Time: 54:51)



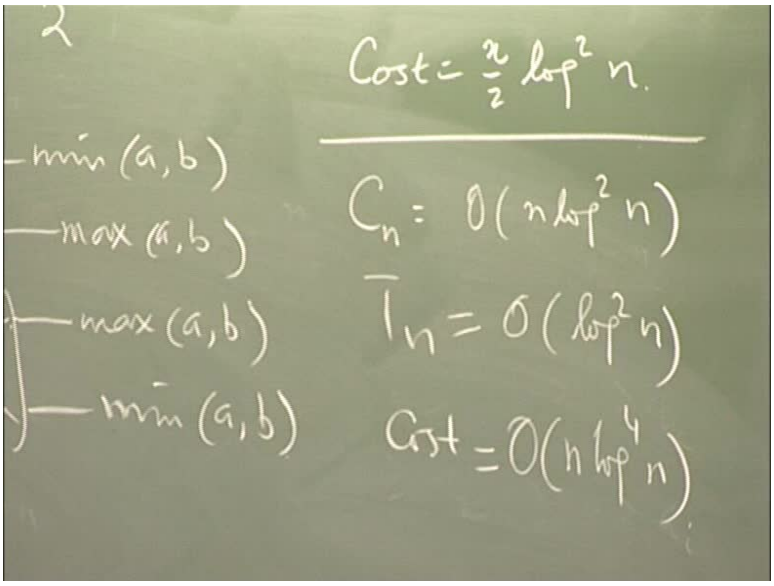

(Refer Slide Time: 55:08)




In the i th stage, 2 to the power n minus i sub parts are there. This is a bitonic sequence is size to the power i. Agreed? So, this is 2 to the power i minus 1 i. Summation over. i equals 1 to log n. There are n minus 2 to the power n minus i sub parts are there. Each part is a size of 2 to the power i, which is a bitonic sequence. To do that, you need the number of comparators i into 2 to the power i minus 1, which has come from here. 2 to the power, yes, this k minus, yes, this 2 to the power k minus i, yes. Agreed?

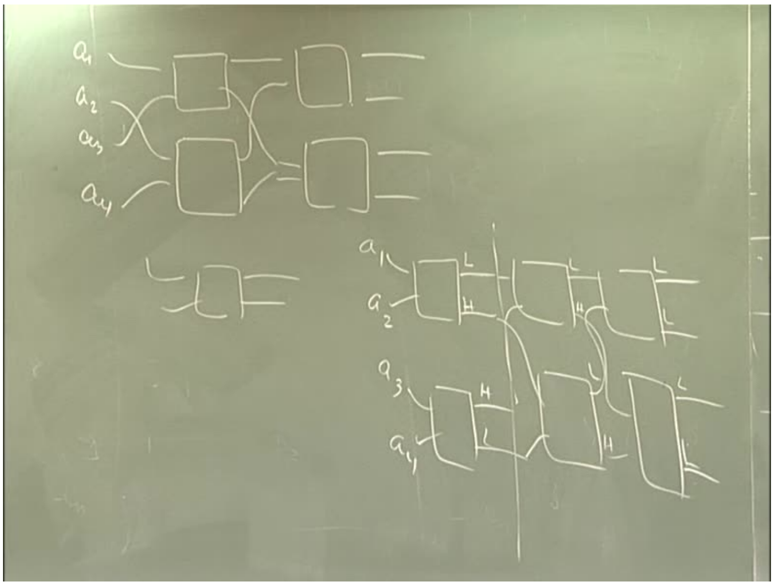
If it is that, this becomes summation over 1 log n 2 to the power k minus 1 i 2 to the power k minus 1 goes out and it is submission over i, which is again log of n log of n plus 1 upon 2, which is order n log square n. So, you can write number of compotators you need order n log square n and number of parallel steps is order log square n.

(Refer Slide Time: 56:39)




So, for sorting n elements, you need, the cost is order n log to the power 4 n.

(Refer Slide Time: 57:17)



So, it is also far away from the cost of optimizing point of the (( )). What should be the basic unit of bitonic? Basically, you will need, suppose you have 4 elements, which forms a bitonic sequence. That is, 2 elements and 2 elements, you need to give the sorting sequence. If you have 4 elements, which forms a bitonic sequence and the structure is like that. Now, if you have to combine both the things for obtaining a sorted, suppose it is an arbitrary element and you want to have the sorting sequence, you can easily find out that. Suppose this is arbitrary one, a 1 a 2 a 3 a 4. So, first you make, this is low and this is high. This is high and this is low. Now, this becomes a bitonic sequence.

Now, in the second phase, you get low low high. So you will get the sorted sequence, right. Now, if you observe or if you compare this with odd even merge techniques, you will find what? If the same number of competitor sure (( )) order wise it may be same, but, you are using more number of comparators. Because, if you observe that was, if you have odd even merge, you first you do odd even merge plus n minus 1. You do not need any comparators here. You need, right, so number of comparators are increasing. Cost wise it is the same order. There is no problem. Moreover, in bitonic merge, you have to keep track of odd even and you have to keep low high high low, all those information, which you do not have to keep for odd even. So, this is all about your bitonic sequence. Now, any questions on this? If there is not, I do not want to start the sorting algorithm.

It is a bitonic sequence. You will realize, for network model, we will discuss how efficiently you can use the bitonic sequence to sort an element, which may not be possible for odd even merge which is present on network models. We will discuss it. Yes, not only that, number of comparators is greater, it is also you have to keep track of how many low high and how many high low comparators you need, right. Did you, could you go through the or could you get the time to open the book and go to volume 3. No. Is there anybody? No. I asked minimum sorting network, minimum number of comparators that type of chapter is in the book, volume 3, chapter 3 (( )) 5.4 or something like that. You please go through that. Sorting network discussion is there. You can you use any one of them. See, here you observe what we need. We started with two elements and two elements. Instead of that, if I use 5 elements, whether it can achieve the goal of sorting efficiently or not. If it is, then, how much? That type of things you can get. Because, what we observed, that if the number of element is odd, then number of comparators take less (( )) to sort.