### algorithm\_week2\_3-20240909

说话人1 00:01  
To finish the whole song without any mistake.

说话人2 00:06  
All right?

说话人1 00:07  
And then you see, according to this design of the pad, if you move 1 foot from up single to down single, it's the longest distance. It takes a lot of energy, four. And then if you move the cases, it's three. If you move center to any path, it's two. And then if you just tap in the original place, you can move your feet foot, then it's one ok so the energy are all given to you. Now, the question is, if I give you a sequence of symbols, how to calculate the best structure,

the symbols are to do up, down, right, up, down, et cetera. I want to calculate, right? And then a very straightforward formulation,

说话人2 01:00  
whether

说话人1 01:01  
is to use di what is didi is the amount of energy or minimum amount of energy you need to spend up to single I this is a natural right up to I single. How much energy do I need to spend in order to be totally correct along the way, right?

And then you compute di plus one,

di but when you compute di plus one, it seems that we meet with some problems, because if you don't know how to get the n plus one from the I what's the reason? I don't know where you think about it.

So somehow you remember where you think about it. So naturally, you need to recall the position of your feet, ijkokjk can be up or down something like this, right? Then the di plus one something, vi plus one to the position of two beats will be equal to the minimum energy of a lot of choices. Right? What are the choices here? The choices can be on your left is up, right of this down, right? And then you move to a new place. How much energy do you need? Right? So basically, it depends on the the history, right? Where are your feet in before step I perform? Then you can calculate how much energy they need to move, my right foot up, on my left, foot up, et cetera. But your previous history can be many, right? You can have many previous history. So the number of branches here could be as many as 5 × 5, 25.

Now in the worst case, I why? Because in the previous step, your foot can be

any combination, right? But you will challenge me. It's not true. Your previous step. One of your feet must be in single aisle, right? So at least I I can fix what's 1 foot of mine. It must be there, right? So your number of states drops from 9:35 ms and then they say, maybe I can use the knowledge. Instead I minus one, and it exclude further options, but it's too complicated. When you write for. You don't want to see some of the difference choices, right? Or different scenarios to deal with this kind of approach to map to do this. I the recursion is not very wise. Here, for this question, what i'm trying to tell you is sometimes you can change your view or angle of looking at the problem.

Just now we say we use ei to represent a number of author, the amount of energy, the minimum amount of energy you need up to single iron. Right? Let's change our perspective a bit. You say, now I use ci to represent how much more energy you need to complete the whole song. What's the minimum amount of energy you need to complete whole song from now on?

说话人2 04:30  
Right?

说话人1 04:32  
So it seems it's only a compliment, right? Just now to say up to step I right? How much energy that mean? Now you say you tell me from step I how much more energy do I need? Right? It should be the same, but actually, it's not right. So if I use this new interpretation, then i'm computing bikk from di plus one.

说话人2 05:00  
So right?

说话人1 05:02  
Because di plus one means from step I plus one on how much energy do you need, right?

And then i'm going to use that one to compute the energy starting from step I

because now the step higher onward, this solution needs more energy. This is the ijk is equal to the minimum of some branches plus something, and

说话人2 05:28  
this plus something. But this is

说话人1 05:32  
pretty good, right? Because now where your feet are is step up.

And then you step I plus one. You don't have many choices.

说话人2 05:43  
Why?

说话人1 05:44  
Because step, I just want the symbol, right? So you wanted to have two options. One option, move your left foot to that symbol. The second option move your right foot to that symbol. Now you only have two branches. Branch number is two. Just now remember the branch number is ten. Now it comes to, just because you changed understanding of the I ws, right? From past to future. That's a very elegant treatment, right? But details i'm going to see that if the agency can try to call something,

说话人2 06:24  
right?

说话人1 06:24  
Any questions about this example? This example is not too difficult, right? Now i'm going to explain the it actually seems not very stable. So I have to put it here again.

The next example i'm going to talk about is called tree model for tree model. It's called comfortable group. But in terminology of graph theory is also called independent set. What's the setting?

Again, this is one place where you need to use knowledge of data structure, but this data structure is just a simple data structure. It's called tree. What does it mean by tree? Trees? Means you have roots. The one root, which is the top node. Then this node has different branches below it and different branches. They do not have connection to each other.

说话人2 07:26  
All right?

说话人1 07:27  
They are independent. And each different branch themselves is also a tree. A so called recursive definition tree is one route together with several trees. You see, this one, it actually has this one tree with a tree, with one root and three branches. The right most branch has corners. Right? Middle branch has one node, and the left branch is one node.

Now is comfortable group. Imagine history, models, relationships among people. If you have an edge between two nodes, now every node is one person. You have one edge between two people, two nodes. That means what? It means that they are enemies if they do not want to be together,

说话人2 08:15  
right?

说话人1 08:16  
Okay, and also go is that now somebody is organizing a party, they say, okay, please invite some of your friends to what? Yeah, invite someone to join the party.

So these are the group of people you can invite, okay? You can invite a subset of people from this victory, then who can invite, then ideally, you will not invite someone who are endings, right?

说话人2 08:40  
That's not good.

说话人1 08:42  
You are trying to invite someone such that between any two of them, there is no match. There is no connection. Clear on the problem. This is called comfortable group in my second.

Okay. Then what I have to do, it definitely is a dynamic following, right? But then because it's in the topic and improving, so it must be done informing. But how to solve it here, opposing a straightforward formulation called du what's u use here? Every note I give it a name ok so du means, if I see that people from the tree, with you as root, how many people can I see that? What's the largest amount of people I can say?

Ok that's the definition of du if I divide du this way, can you draw any relationship between du and the children below it? We say this is the children of the 3 months, for example, this is v one. Is v two, v three. So can you derive any relationship between du and this dv one, dv two, dv three? You're trying to solve some problems. Sometimes. There's no answer. Remember this one after some time,

说话人2 10:27  
missing cynical

说话人1 10:28  
to find relation between eu and ed one, ed three, ed two is.

And then you can figure out why why is it difficult. I can tell you why you cannot find a relation between du and du and dd to dd relative means. How many people can see that? Right? The largest amount of people can see that within the tree is us rooted right wholesale.

Then dd one is what if you want me is how many people can select here, right? Dd this is dd two. Why is it difficult to make the money? Any idea? What's the key idea? What's the key challenge .? Can I say du equals dv one plus dv two plus dv three? Can

说话人2 12:10  
I say that?

说话人1 12:13  
No, why not? Do you want me to be? May have x different? What's the key .? Do you see? You say? Now, first of all, the solution for dod one, dd two, dd three, they have no dimensions. They can. Delta is separate. It's okay. But who was the key trouble maker? Yeah, he was the key trouble maker, right? But because what can I say that you actually don't know, because if we didn't see that any of them, you want me to be three. I can see that you, right? But if you see that one of them, I cannot see that you.

Okay, so that means actually, your information in the cu is not sufficient. Instead, you need to tell people whether the route is selective or not. If you tell people whether the route is expected or not, then it's very easy to make up the or combine decision,

说话人2 13:22  
right?

说话人1 13:23  
That means the state here should not be du it should be d u zero and du one.

Du zero means when I do not see that u how many people do you see that industry in this industry? Right? And the one is, if I study, right, how many altogether you can see that in history? If you do this,

说话人2 13:52  
then

说话人1 13:53  
you can compute du one as what? I'm going to be 111 sub problem, the other one, we can, again, do it as an exercise du one equals one. That's easy one. That's the easiest one, because that formula is short. Yeah, very good. It's dv 1100, very good, plus dv v two zero, right? Then plus d is reason. Right? Because if you see iku so none of the children can be set out. So for some problem, becoming about the zero version, zero is not together.

Good. And then similarly, you can compute du zero.

Now, I believe that I don't need to do that here, although they are different, that we have. All

说话人2 15:02  
right.

说话人1 15:03  
This is the second new dp problem we analyze here. Okay? The next one, i'm not going to talk about here, but I want to tell you some background on this. So actually, this one is a the first research problem I studied after I enjoy it to you. It's a very interesting research problem with relationship between the code loss in the program.

So basically, code blocks can be modeled by a tree. Okay? Also, this kind of structure. I first entered this block, and then with some condition, if this happens, I go here, else I go here. I have some branches. And then this version says, if i'm able to put some code blocks in the

说话人2 15:57  
cache,

说话人1 15:59  
then usually I can speed up the process. Sometimes ii go to some certain parts of the execution. I can save my time reading blocks, because they are in cash. And the question is, which loss you put in cash so that the worst case execution time is as small as possible. Now i'm not going to talk about the further details because of the time limits today, but i'm going to come back to that problem sometime later in the first 44 weeks. So basically, this is, again, aa question. I like a lot. The reason is it combines the greedy algorithm for some special case, and pp algorithm for some general case. And also it involves some it refinement of the dynamic programming for the general trip.

So I will talk about this problem later, but today I think our focus is Economical here, right? Okay? The remaining time i'm going to talk about the last standard approach of everything. Design is called divide and conquer. I believe that some of you here, we raise the issue of advanced conquer, so now we really come to, okay, what is do I come here? His philosophy is similar sdp actually, or as similar recursion. Development congress says, I split the big problem into smaller problems and solve the smaller problems separately, and then combine a solution together to become a solution for the big problem ok and then they say, how do I solve the smaller problem? Easy, quick question.

A small problem can be split in third smaller problem. Any salt

说话人2 17:49  
decision?

说话人1 17:50  
Actually, the line complement is another how to say formal statements for recursion.

Okay? So first of all, let me do some more about exercise, okay? So there's very classic areas of problem called selection problem. I believe everyone knows sorting. I thought the numbers and sort of numbers from largest to smallest, right? I but here we are looking for a simple version. We are not asking to solve the numbers. He said, i'm asking you to see that the case by the sum, for example, as you see at the largest number, very good. Largest number is very easy, right? Isn't it? And then return done over time done. How about second largest? Do you know, the best algorithm for the second largest or not just together with second largest? Again, it's out of the scope for today's lecture. The largest plus second largest, the best time is n plus log n no, sorry, very interested. We can go to related message ok not today's. So now, in general, if I want you to find the case largest number, what are you doing? Any suggestion? Maybe the slowest error and everything that gives you a lot of information which are not necessary.

And then it can also at the same time give you the case understand what is that? It's very easy. So the last okay, I always see surprises in the answer. That's very good. But remember, ii the slowest algorithm, which gives you too much information. But at the same time, it also tells you the case largest number. What is

说话人2 20:11  
that?

说话人1 20:17  
Actually, I only give you the hints at the very beginning, inversion, or no, sorry, not at the very beginning of this lecture of this course. It's at the very beginning of this topic. So actually, it's just one.

说话人2 20:45  
Right? I

说话人1 20:46  
told you, ii I mentioned sorting, right?

Actually, the first element is solved. I saw all the numbers, and then it's very easy to find a case like this, but the running time is very big. It is high. What's the running time here? Definitely. Because it's quick sorting out with this n log n by comparison. It's n log n this one. Ii I remember which major you are in. You must know sorting how many of sorting? No, I do not know. No sorting president. Yeah. So how many of you have learned? Salty? Raise your hand. Okay, still not what. Anyway, so you have many kinds of emerge sorts, fixed sort, et cetera. So many good sorting algorithms. They give you n log n running time. This is a service man. Because as I said, it tells you too much information. I don't need that much. So now comes the second paragraph. The second there some. It's also not difficult to think about the second of the test.

Anyway, you only need me to tell you the case largest number. So it does not make sense for me to know or want to keep many more numbers. So I just copy the first k numbers into my memory, first key numbers, right? Now, these key and first key numbers may not be a final solution, because in the future, there could be larger numbers coming in, right? Ii pretend those numbers come in one by one. All right? And then I take the first thing numbers in my memory. And then for any future number, it comes in. I try to see whether it's larger than the key numbers I saw. If I will replace some number in the k number and kick out smallest

to do this efficiently, I won't maintain an order of these statements.

I keep it from largest to smallest. Then for every new number coming, I I put the new number in the right place and kick out the last one.

说话人2 23:00  
Right?

说话人1 23:01  
Okay. So what I do here, it's just something like this. For example, I have a 65, 21. I'm going to take the 4th largest number. I have four numbers here, and then I have three coming in. I just put three in and kick one out, right? Ok so in order to put three in the right place, how much time is it? Yeah, very good. Log k you have key numbers altogether. It's log k time, binary search, right? Binary search to find the right place.

说话人2 23:47  
Okay?

说话人1 23:48  
So lucky time for every new number. How many numbers do you have altogether? It just end them inside. So over running time here will be to be accurate. It's just n minus k times log k plus k log ok this is sorting of the first k numbers, and this is the remaining numbers coming in. And this log k is the operation needed for each new number coming in. Altogether, this running time is just n block taylor. Ok n block t definitely is better than n block n because k is smaller than n good.

Now, the second approach. Now, these are only warmer, not the final answer. What's our final answer? Our final answers looks like this. So we try to recall quite. So what is fixed a bunch of numbers?

说话人2 24:58  
And

说话人1 24:59  
see that one number is a reference value for pivot, that one number as a reference and divide the remaining numbers into 2 groups, 1 group smaller than this number, the other group larger than this one.

This is a so called petition process. Just like this. If I have these numbers, like 65321, so maybe I choose five as a reference number, then I divide the number into five here, and then 321 on the left, and six on the right. I can use one value to split the remaining numbers into 2 groups, one group smaller than the black.

说话人2 25:41  
All right.

说话人1 25:42  
So if you are very lucky, the reference then took, you repeat, is just the case largest number. Then you stop. This value is your answer, right? I'm very good. If this value is not case modest, the index is larger than what we do. So its position or from largest to the smallest position, it's larger than k but that means this reference that is too small, right? I wanted big one. So you will search in the larger part of the 2 groups.

说话人2 26:15  
Right?

说话人1 26:19  
Trnk that means this reference site is too small. The rank is too low.

Then I will search, for the case, largest number in the in the group of big numbers, right? A certain case, largest number in a group of big numbers. Right? Now, if p this reference, p rank is smaller, which means this area is too big, right? It's very ahead of in the sequence. Then you should search, for the case, not remember in the smaller part of the group.

说话人2 26:51  
Right?

说话人1 26:52  
Smaller part. So in other words, if i'm finding the third largest number, right? And finding the third largest number in the sequence, now, by reference is five, there's two, right? Because 65, and then three small numbers, right? So by rent in the whole sequence is too big, right? I'm looking for the third, sorry? Yeah, i'm looking for 1/3, right? I'm looking for 1/3. So you're going to look for some number that this number in the smaller part of this sequence, which position in that smaller part, the position is this k minus p minus one. I don't know how to explain why it is. This case. It's very easy to calculate.

All right? Anyway, if my reference is too big, please go to the smaller part search. If my reference is too small, please go to big part search. Right? Anyway, the problem now becomes small.

说话人2 27:52  
Right?

说话人1 27:53  
Now, the important thing here is how to analyze that, or how do you like good areas, how to split, how to choose the reference set? Now, this part, I think, will use up the remaining candidates. So a very straightforward way to select the pivot for the reference value is to see that some value, which is somewhat in the middle, because we do not want the extreme value. For example, if you have your reference is the biggest one, it doesn't help you, right? Then you still have a very big problem. That's not good. So we are trying to find some reference in which is somewhere in it.

Now, one way is to see that the so called medium of the first 2/3 of the numbers,

median of the first two steps. So I think that i'm going to use some whiteboard. Eventually you could have missing space. Altogether, you have n numbers, and then I have 1/3 n here, 1/3 n here, and 1/3 n here. So for the first 2/3 of n numbers, I take the medium number. What is medium? It just middle, right? Middle number, middle number. I take the middle number of the first 2/3. I find a number here, and this number is able to split the holes, all the numbers in 2 groups. Right?

Now, what can you say about these 2 groups? How many numbers are these 2 groups now? Because this representative is a median of this set, right? You can see that the number here must be larger than one per game. The numbers here.

说话人2 29:56  
Is it correct?

说话人1 29:58  
Using this reference set, when I split the numbers? The left part and the right part, each part must contain at least 1/3 and numbers.

说话人2 30:08  
Why? But

说话人1 30:13  
because i'm using meeting for the first 2/3, meeting is middle, right? Then there was the about 1/3 number on the left of me, right? About 1/3 number on the right of me. Right? These are already fulfilling this requirement, right? No matter where those numbers go, it doesn't. Even if you go to the extreme right or to the left or to the right, it doesn't matter. Right? Because my left part and right part already have sufficient supporters. They already have answered that.

Then this using this reference value to split, you can guarantee each part is less than or equal to 230.

说话人2 30:56  
Is

说话人1 30:59  
that right? Because you have at least 1/3 number, which means the other side, at most, 2/3. Right?

So your regression is very successful, because no matter which part you go to, you have a size, at most, 2/3. And right. Now, when you check the code box, every time you draw this, it will have any, for example, I don't know how to solve it anyway.

When you see this repressive formula, you will see what did you do here? T two, n over three. There are two, tn over three. There are 2 copies, right? So there are,

说话人2 31:47  
you see,

说话人1 31:51  
can not. There's two here, right? There's a two here. But this too does not mean a sequel to. It means two different meaning of t two analysts. The first t two no of three means the time to find the medium in the first 2/3 of n numbers. Because finding median is a special case of finding the case largest number. Here we use tn to represent the time to find the case largest number.

The time tn this process, right, can be split to step one, finding the median of the first 2/3 n numbers. This process takes t two, n over three. This is the first two, at two anniversary. Second, because i'm going to use the reference number to split the numbers of 2 groups. This split process takes all end time, because you need to check every number smaller than reference or larger than reference, right? You do check. And that's why you need all in time to complete the process of competition. Right? Now, after you do petition, we know that you will go to one side to do recursion. I say either go here to to a further communication, finding certain number in that group, or you go here to find a certain number in that group.

But anyway, no matter which group we go to, The size of the group is only two and over 3, right? That comes the second, two and three. Remember, the first two m over three, t two m over three is the time to find the meaning of the first 2/3 number. And the second two number three is because go to the sub problem to recursive, they compute a certain number. Using this formula, you can see the overall running time now becomes n to the power of 1 . 7, which is better than n squared. Remember, where is n squared? N squared is the previous page if you simply do this, just like it. So the worst case is Pennsylvania.

Now, we are able to improve it to enter the power of 1 . 7, right?

说话人2 34:14  
Okay?

说话人1 34:15  
But our ultimate goal is to improve it to linear, which is on this is a very challenging task, but still, there is a a group of scientists who are very smart. They figure out the following way. By doing this, they are able to do linear time. What's this approach? This approach is the following. It says, now I split all the numbers into a group of five.

Every group has five numbers, and then I find the medium of these every

说话人2 34:51  
group.

说话人1 34:52  
I find the middle number for everybody.

说话人2 34:55  
Okay? Step up.

说话人1 34:59  
Now, this step takes constant time or all end time, right? Because you only have handle the table of five or the 5 groups, right? Every group take a little number is very fast. So it's just the em

说话人2 35:15  
there it is.

说话人1 35:15  
Now, next step, among these mediums, I compute the medium. Again. I compute the medium of done. So now you have handled the five numbers, right? Among these n five n over five numbers, I picked their meaning. The time to pick their meaning is tn over five, because they have n angle five numbers. I pick a medium that's just a calling for this problem. This is tn over five times.

Now, after I pick their medium, now you can divide the groups into two sites. One, this suppose is a median. This is a median of the n over five numbers. Now there are some people on the left of him, some people on the right of him. Right?

说话人2 36:03  
Now,

说话人1 36:04  
using this number to do split, this split is pretty balanced. You can imagine because all these numbers smaller than this. So everything smaller than their medium, they are also smaller than this, right? Every group, the medium is already smaller than this final medium, and then anything smaller than respective medium, they are also smaller. So basically, this region is is totally small. This region is totally small. Then this star. And then similarly on the right, you have another region, which is, sorry, I should not use less than. I should use this one 0 k so there's another region of numbers which are definitely larger than this, meaning. It's similar as to what we discussed before.

So basically, on each group, on each side, you can guarantee you have at least certain number of numbers. Then this split is the so called 30, 70 split, which means on every side, there are at least 30 % of numbers, which means on the other side, there are, at most, 70 % numbers.

说话人2 37:16  
All right.

说话人1 37:17  
This kind of split guarantees something, which means when you recurse, when do recursive the smaller problem, you only suffer this kind of running time, t seven, n over ten running time, because that's the largest size of the handle, right? In the next step. So you're over running time is finding medium of the n over five numbers, and then do position, which is the win. And finally recurse into a big problem of AA relatively large sub problem, which is seven out of ten.

Now, adding these three together, that will be a bound of your TM you can see, actually, the coefficient of these two terms, 11 / 5 and 7 / 10, they add up to a value smaller than one. And then you can guess, suppose this CN is less than equal to kn then you can prove it. You see, basically, this regression just tell you the overall tn is just the way this meaning.

Now, this will be the most difficult example i've ever explained in that in this course. So it's totally fine. Then you cannot understand on the first page. Okay? But if you are interested, you can go to the related literature, either on internet or textbooks to find a reasoning. So this one is really, really smart because it can bring down the complex convenient, which is really amazing, but the coefficient makes a difference here.

All right. So I guess today I will just stop here, and I will stay behind for a couple of minutes if you have questions. Welcome to us. By the way, I forgot to do the survey as a promise who remember what the survey is. It's

说话人2 39:17  
a normal question.

说话人1 39:20  
I'm going to do a survey at the end of the or voting or something. What's that question? Please? Only official students can answer this question. What's that question? 54321

that you lose this .. The question is, whether this course is more difficult than most people cost. You have already taken. The first of all,

说话人2 40:02  
okay.

说话人1 40:03  
If you think this one is more difficult compared to any of the previous posts where they have no one, then I'm very, aha, but that's good. Anyway, it's a one, started a six, but that means the same course before me. Probably could be more difficult. But anyway, so you will see more details later on. Yeah.