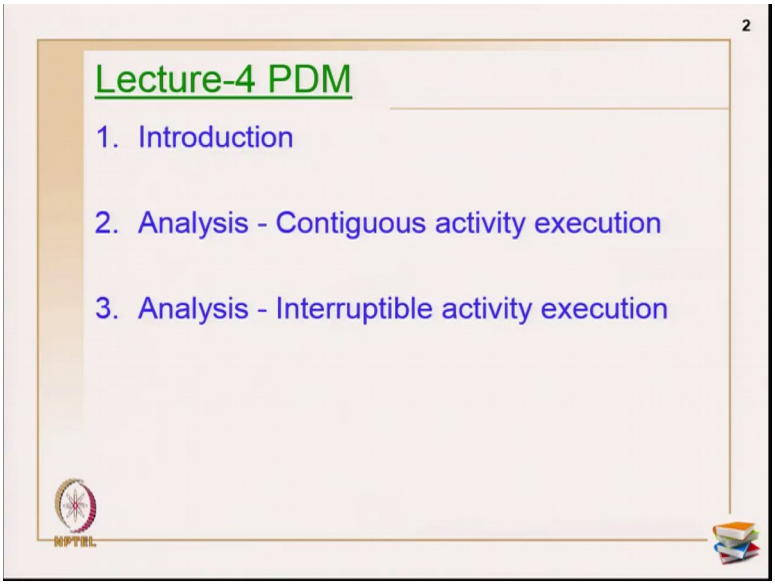
Scheduling Techniques in Projects
Dr. J. Uma Maheswari
Department of Civil Engineering
Indian Institute of Technology, Delhi

Lecture – 04
Precedence Diagramming Method

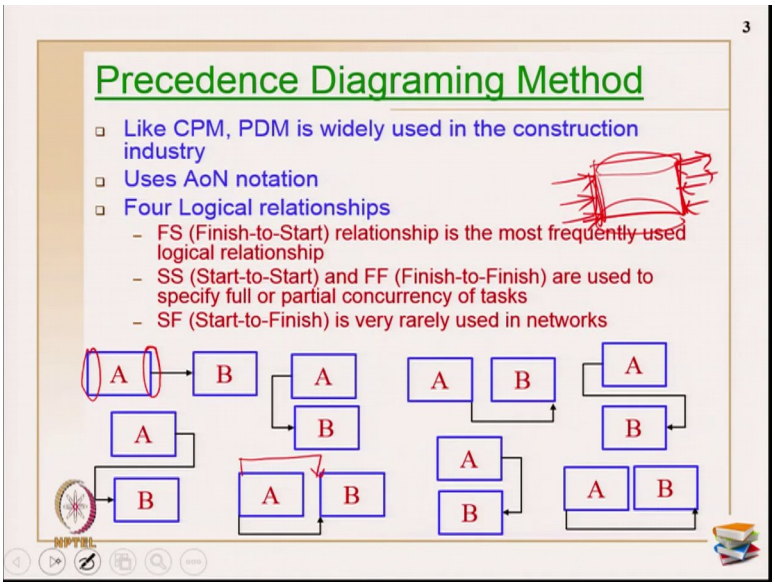
So, the course outline or primarily the lecture outline for today's class will be like this.

(Refer Slide Time: 00:23)



I will introduce, what is PDM, how to draw a PDM and so on. And then for network analysis, we primarily have two different methods; one is contiguous activity execution and the other one is interruptible activity execution. In the sense, when an activity starts; in the first case it does not stop in the middle and it executes till the end, in the second case it can have a break in the middle of the execution as well. So, we will see how the analysis goes.

(Refer Slide Time: 00:53)



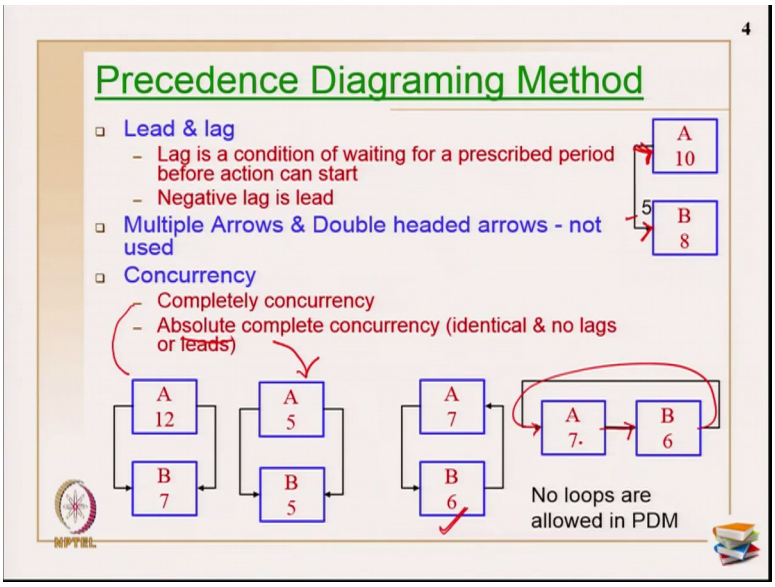
And now, what is PDM? So like CPM, PDM is widely used in construction industry. We have seen earlier class, what is CPM; it is critical path method and this is also very widely used and it uses only the AoN notation, which is nothing but Activity on Node formation only. Now, what is the specific difference between CPM and PDM is, the presence of four logical relationships and the lead and lag, you will see in the next slide. So, what is the four logical relationships? Number one is finish to start relationship. As you can see here, this is the most widely used relationship in CPM network, CPM, PERT and so on.

So, this is nothing, but when an activity is executed completely then the next activity starts, that is the meaning of after finishing the other activity is starting. So, primarily it defines the relationship between any two activities. And how it is represented in a PDM? As I told you earlier, the nodes can be circles or rectangles and generally we use circles for CPM and rectangles for PDM, ok. So, I have represented nodes and as I told this is an AoN representation, so activity on nodes. So, I can use this format or I can use this format, so both are fine, ok. Then, the next relationship is start-start relationship and finish-finish. Generally the start-start and finish-finish are used together to explain the partial concurrency of activities, ok. This is start-start, after A starts, B should start ok; that is what I am representing here. You can also use this, you can also represent like this ok.

So, any form is fine and finish-finish; so this is a finish-finish. After A is completely done, B should be completed and the same thing I have represented when it is arranged in the vertical order. Next is start-finish relationship, which is these two figures. So, this explains after, when activity A start B should finish which is very rarely used in networks and it is not that widely used ok. So, these are the four relationships we have in PDM networks, ok.

Now as I have explained you here, so always the start part of an activity is only this and the finish is only this. So, if I want to draw a network ok, I can use my arrows anyway from here ok, it can end up here. So, this is primarily the start portion and when I want to end it, this primarily called the finish portion. So, I can do like this or I can start from here like this, ok. And never you should use this segment of the whole box, anyway the meaning of that you will understand later.

(Refer Slide Time: 03:41)



So, next is lead and lag. As I told you the two main functionalities in PDM is, the other types of relationships and the lead and lag, which is not available in CPM. So, what is a lag? Lag is a condition of waiting for a prescribed period before action can start. So, I have given an example here. So, A is an activity, B is another activity; this takes 10 days and this takes 8 days. So, this says; it is a start-start relationship. So as soon as A starts, B can start. But I have given a condition called 5. So, as soon as A starts after 5 days only B can start and I have to wait till the period is over, ok. And the negative condition is

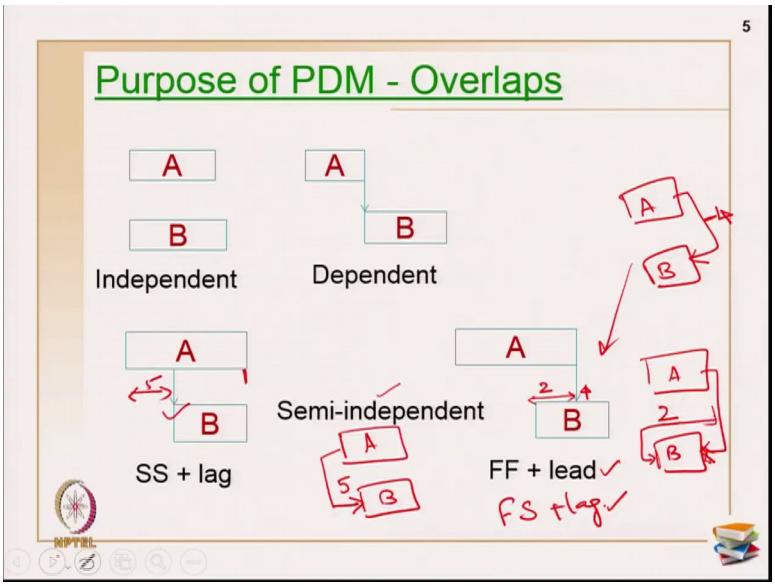
primarily called as lead, which implies suppose I am writing it as minus 5; as soon as A start, so B should have started at least 5 days in advance, ok. That is a meaning of negative lag, which is otherwise called as lead, ok.

Now, multiple arrows and double headed arrows ok, primarily those are all not allowed. The meaning of both is same. So, primarily I am not supposed to have two arrows in the same arrowheads ok, which creates confusion when you are doing calculations, which is generally not used. As I have explained in the last slide so, start-start, finish-finish are generally used in combination in order to explain concurrency or partial concurrency of activities, ok. There are some terminologies in the concurrency. So, one is called complete concurrency; other one is called absolute complete concurrency, ok. So, this complete concurrency is explained like this. I have an activity A, duration is 12; B, duration is 7; as soon as A starts, B also starts; as soon as A finishes B also finish.

The logic is there is an activity with a larger duration and there is an activity with a smaller duration; and the activity with the smaller duration is completely executed during the course of an activity with the larger duration. It can have lead, lags whatever you want you can keep it in complete concurrency. This is primarily an example of complete concurrency. This case is absolute complete concurrency as the name absolute implies the duration should be same; there is no lead, lag. So, exactly in the time frame of one activity, the other activity is executed. It is primarily called complete, absolute complete concurrency, ok.

Now next, in PDM loops are not allowed. So, what do you mean by loops? So, some examples I have brought in up here. So, this is a start-start relationship, this is a finish-finish relationship; but it is not like these two cases ok. In this case what happens, it just goes in a cycle wise and these are all treated as a loops. In the computer programs, this logic is generally not allowed. Now what about here, this is primarily here finish-start relationship. This is a finish-to-start relationship again. So, two finish-start relationships generally transform as a loop also. If you see here, this also is traversing like a cycle. So, these relationships are generally not allowed in PDM. So, when you are working on PDM, you have to see with combination of few activities that you are not landing up in any loops, because loops are not allowed in PDM networks, ok.

(Refer Slide Time: 07:03)



So, what is the purpose of PDM and why do you want to have these relationships. Why cannot I go ahead just with finish-start and why are we complicating the networks ok. There is a purpose behind. So far in CPM networks we have understood, the there are two types of relationships; one is a independent relationship, where in the two activity does not have any relationship with each other, and the other category is primarily called dependent relationship, wherein one activity is dependent on the other. For example, the successor of an activity generally is dependent on the predecessor completion and so on; that is primarily called dependent relationship.

Now, there is something else which is called semi-independent, ok. I need not have a conventional finish-start, if I able to overlap the activities, ok. So, instead of waiting till the end, why cannot I start my second activity, if the outputs for the first activity is really generated from the same, ok. That is primarily represented a semi-independent and this is not a forced overlapping; this is a natural overlap. There is an input required for B and it can come somewhere not at the end, it can come somewhere in the mid execution of the whole activity. Then I can start my B at that point of time, that is primarily called semi-independent, two cases I have shown here.

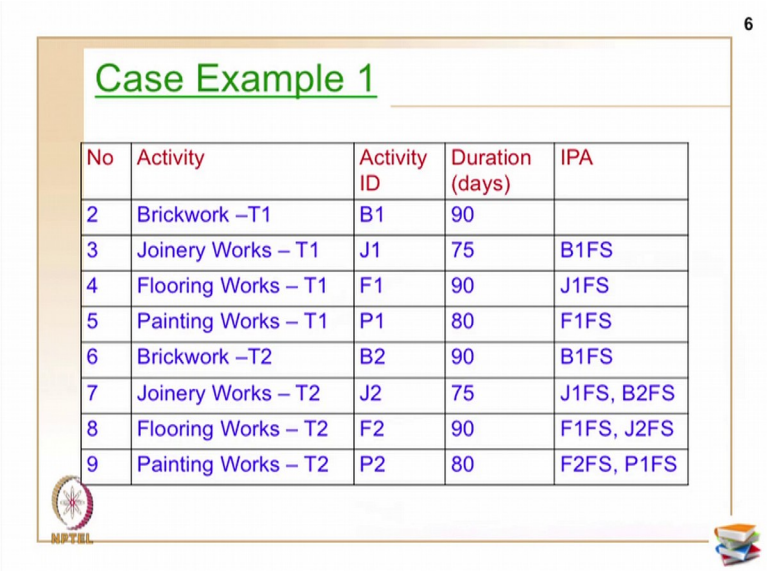
Now, how do you represent this on PDM is, this can be represented with the start-start plus a lag. So, if you want I can show this, so I having an activity A and I having an activity B. So, I can show this with a lag, so maybe 5 or something, if this duration is

something as 5, then what I can do is after activity A starts after 5 days I can start my activity B. I can easily represent this network on a PDM notation. The same thing I can show it here. So, this is primarily by using a finish-finish relationship plus lead values. So, I am having an activity A and this is my activity B and this I can use a finish-finish relationship, this I can use as a finish-start relationship and I can put a lag here ok.

So, this can be finish-start plus lag ok. In the case of lag, I can explain how much time of B has to be executed and then B can start or I can do is, I can have the same activity A, I can have this activity B and I can put this finish-finish; but I can show you a lead ok. So, minus some number maybe 4, here I am representing maybe 2. So, this can be 2 and this can be 4. So, before 4 days of B completion, you have to activity A has to be completed, that is a meaning of this. And this is after my start-start relationship, 2 days after then I can start my B. So, either I can use finish-start plus lag or I can use finish-finish plus lead. Both the cases are available here and both options are available.

So, it is primarily based on, what is the information or what is the relationship I am having between A and B and what type of functionalities I can use. So, with the combination of the relationships, easily you can show overlaps in PDM. That is a one of the main purpose of PDM networks.

(Refer Slide Time: 10:23)

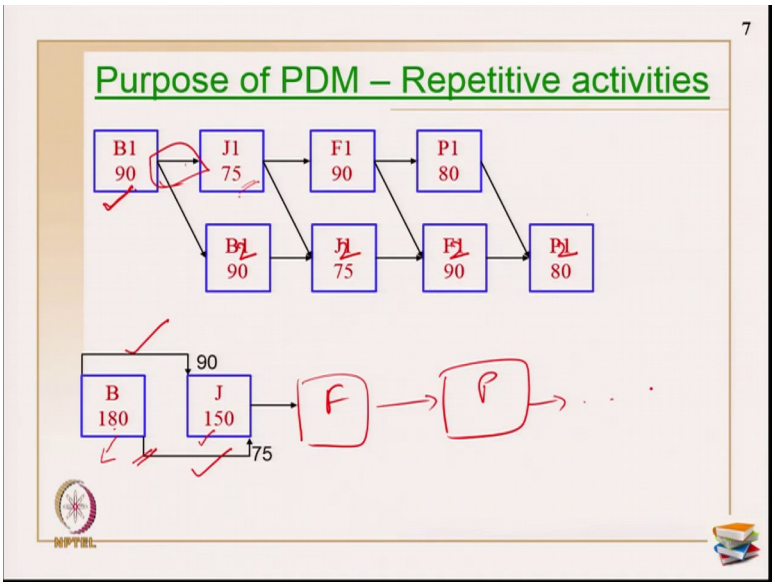


Now, I will just show an other example just to showcase the other purpose of PDM also. This is an example as the; this is the just a trimmed example from a bigger example and

how do I read this example. There are primarily three towers ok, the real example had three towers, tower in the sense it was an apartment construction and the three towers had series of activities like brickwork, joinery, floor, painting and so on and they had series of resources and etc.

So, I just used a short form for all these activities. So, for all the tower ones, I used 1 and for all the tower two's, I used 2 here ok, that you can see with the last number and the B represents for Brickwork, J for Joinery, F for Flooring, P for Painting, same thing I have used. And duration for all these activities I assumed based on the data I have obtained, 90, 75 and so on, and I repeated the same values for the 2nd tower, 3rd tower, because it was exactly the same towers ok. And the relationship between all these activities are also marked here ok

(Refer Slide Time: 11:31)



Now, how do I represent this? Suppose if you take a conventional CPM network so, how do I represent. After brickwork in tower 1, I have to do joinery in tower 1 and same time I can start my brickwork in tower 2, sorry so, this is a mistake. So, in tower 2 also we can start, ok.

Now, same thing after my joinery work in tower 1 is done, so, after the brickwork in tower 2 is also done, I can start my joinery in tower 2 and so on. That is how this activity representation is generally done. Suppose if I want to represent this on a PDM network. This creates little boring when you are having, let us assume 50 activities 10 times

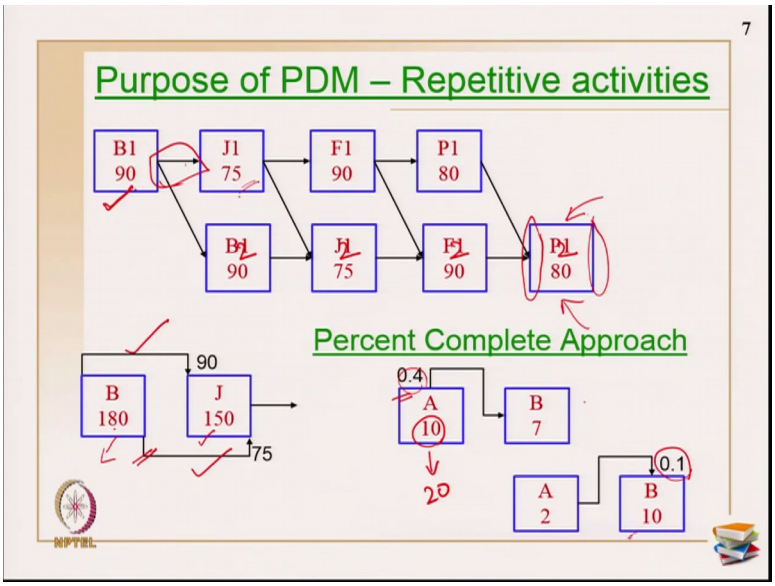
repetition; then it looks really boring for somebody to really see. So, how do I make it interesting in PDM? I can have a brickwork activity here. This is a joinery activity and, I am just showing only two cases here. Total duration of brickwork is now 180 because of 90 and 90, total time taken for joinery is 75 and 75 fo r two towers; I am keeping it as 150.

So, how do I now fix the relationships; so, all these relationships I have, how do I fix it? So, what I am going to do is, after the brickwork started 90 days after the brickwork, joinery should start. So, this will cover my brickwork, after the brickwork this logic is primarily covered. After the brickwork 90 days later, brick work takes generally 90 days; then the joinery starts and this duration is given here with the lag ok. Now same case; suppose if I do not put a finish-finish, I can do my brickwork and joinery may be also in parallel which is not acceptable in this particular case.

So, what I have to do, I have to finish my brickwork at least before 75 days. So, the 75 days is primarily coming from my joinery. So, which implies which is also settling with my finish-finish; so brickwork is fully completed and the last joinery maybe there is a 75 days of lag in order to finish my joinery. So, once I am fixing my start-start relationship and finish-finish relationship, the complete representation on both the activities is really done and this way you may have to extend for flooring, you may have to extend for painting and so on ok. And whatever may be the other activities you may have to extend, ok.

So, this is a main purpose of; the second main purpose of working on a PDM network. First primarily to show overlaps and the next we can also show repetition activities in a very nice way with the help of the other types of relationships which we have in PDM, ok.

(Refer Slide Time: 14:25)



Now, then I have also been telling you always you should use the ends of the rectangle alone for notations and there is a different meaning altogether, if you are using some arrow marks from the either of the horizontal segments of the rectangle. As I have seen it here.

So, this I have an activity A duration is 10, I have an activity B duration is 7 and this arrow primarily is coming from the mid portion in a horizontal segment of activity A and there is a 0.4 return. This implies after 40 percent of completion of A, I have to start my B because it is connecting with the start relationship ok; that is the meaning of this. So, after 40 percent in the sense it implies; after 4 days of A, I can start my B; that is what is represented here. Suppose if A is duration is extending maybe to 20, then accordingly this start of this activity B also you can exchange, you can vary ok. But what happens if you do not use this percent complete approach, what happens and if I just use start-start plus 4 then whatever if the duration of activity also changes; you may still go with the deterministic value of 4 in any case, that you can avoid when you are using a percent complete approach. The same thing for finish also I can do.

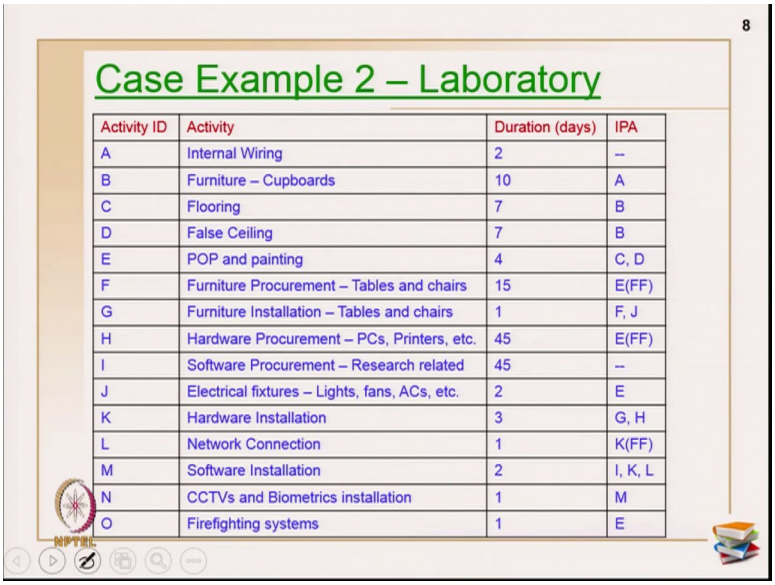
So, I am having an activity A of duration 2, I am having an activity B of duration 10. So, once A is completed, then there is a finish notation to B and that is governed by 0.1 which implies; so, before that 1 percent of B completion I have to finish my A ok. So, 10 days is a duration of B so, 1 percent completion is actually 1 day. So, on the 9th day of

B, so A has to be completed on the 9th day of B; that is a meaning of representing this. Suppose if the duration of A changes or B changes accordingly, the values also change and this is an other usage on PDM network.

So, I have shown you what is a purpose of going with a PDM network compared to the conventional CPM is fair enough and it works well in all situations. But there are situations we wanted like overlaps, repetitions or percent completes, then we may have to go with the PDM and it works well with the other types of relationships and with the lead, lag combinations also, ok.

Now, before coming to that I said there is a prescribed time of waiting and so on; we have very nice examples for all these. For example, in concreting, curing is a very critical task ok; and instead of showing curing as one activity and then connecting it with the finish-start relationship and so on. If you add the duration for curing then with the lag ok; then that shows that you know it is taken care of and it is really understood by the experts as to you have to leave 10 days for curing and so on, that is understood. So, that is the meaning of all these relationships on the lead, lag combinations, ok.

(Refer Slide Time: 17:31)



So now, you understood how to represent a PDM network and what are the situations under which you can develop a PDM network to really get the benefits of the PDM, so all these we have seen so far. Now, how do I get the relationships and where are these relationships coming in and duration for each activity I have shown in the last class on

CPM, how do we get the durations for each of these activities, ok. So, in this class we will see, how do I get the relationships and how are these relationships coming in; little you would have understood with the examples I have shown right now, but more detail we will see.

This is a running example which I am taking from the first class; this is on developing a computer laboratory, computational laboratory ok. Now after 1 or 2 iterations just to develop and schedule. So, this was a real schedules which were developed after all decisions were made, as to cupboards should be made by carpenters customized in the lab and then furnitures should also be customized with the design, drawings and so on and tasks to be made by the carpenters. So, like this several decisions were taken and the durations also had come with that, some of the relationships also come with the help of the decisions you made on each of the activities. For example, if the furniture has to be ordered then the predecessors, successors would have been different, this furniture's has to be customized, sometime's the relationships also change with these predecessors and successors, ok.

Now, these are the list of activities for example; internal wiring that was a first activity that was planned, after the room was partially ready. So, internal wiring generally it takes 2 days, so that all the wirings are hidden inside the rooms. And then next to furniture cupboards because that was like customized option for us and to know what is a height and to fix the full length of the cupboards ok. So, the cupboard was planned till the ceiling height and it was planned to be done first.

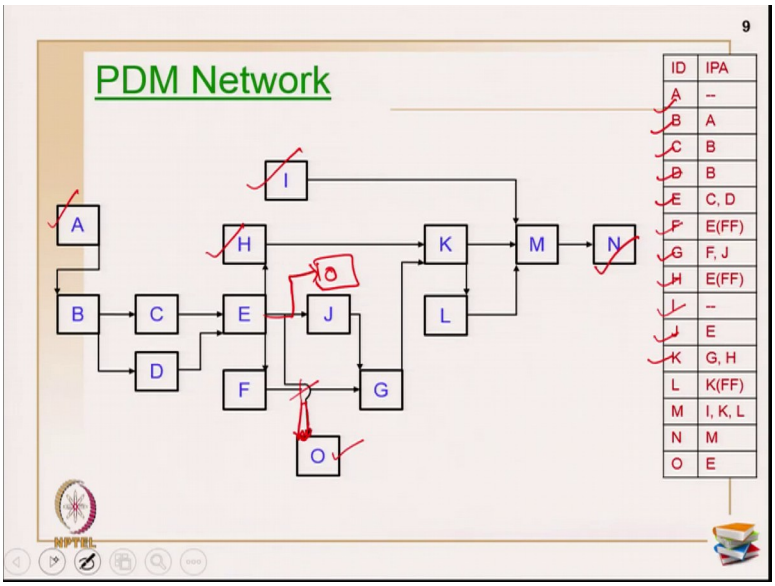
So, after the wiring was done all along the room, then the cupboards were fixed, so that the cupboards were like; the cupboards were planned to be fixed. Then flooring was done and simultaneously false ceiling was also done, because they were not dependent on each other and that could have been done in parallel. Then after finishing the flooring and false ceiling in the remaining places, POP and painting was done on the same, so which took 4 days. Then furniture procurement for tables and chairs were done and here this is like a parallel task; and what is given here is this should be finished as soon as the POP and painting are done, as soon as the room is ready the plaster of Paris and the painting is completely done, then the table chairs can come into the room ok; that was because we need a place to store all these inventory.

The next installation, primarily the customizing and assembling of the tables and chairs were planned. So, that actually happens with the predecessor on procurement and also with the electrical fixtures on, because tables cannot be moved, because we need power points for all the tables, because we are planning to use a computer there. So, that is where we had a link with the predecessors with electrical fixtures. Then hardware procurement on PC's and computers, for which we need the POP and painting because once the room is completely ready then only we can bring in the equipment. And then software procurement; so software can go in parallel, but the software procurement can go in parallel in order to get permissions, approvals (Refer Time: 21:14) quotations and so on.

But the installation if you see it depends on the room has to be done, initial machine installations all has to be done, then only the software procurement also has to be done for the installation to happen that is what you have I, K and L as a relationship ok. So, that is how the relationship starts filling up, after you make all the decisions on the activities. So, the logic is, let me go in little backwards the logic is first you have to go with the levels, breakdown the project, as and when you start breaking down and moving ahead you get more and more information on the assumptions, what has to be done; the real scope of the entire work on each activity gets defined and then you go on in the levels, ok.

So, at one stage all decisions on all of the items would have been done, then the network schedule is generally prepared, ok. So, this is primarily done after the complete example, after the complete what is it called a laboratory setting up is done. So, I may have a real clean network schedules done, but it does not happen in practice when you are taking up a live project ok. But this is only to showcase how to do the example, so I have used this example. Now for the same example I have this, for me I need only the ID's and the IPA's. So, I am, because I want to show you only how to make the PDM network ok.

(Refer Slide Time: 22:37)



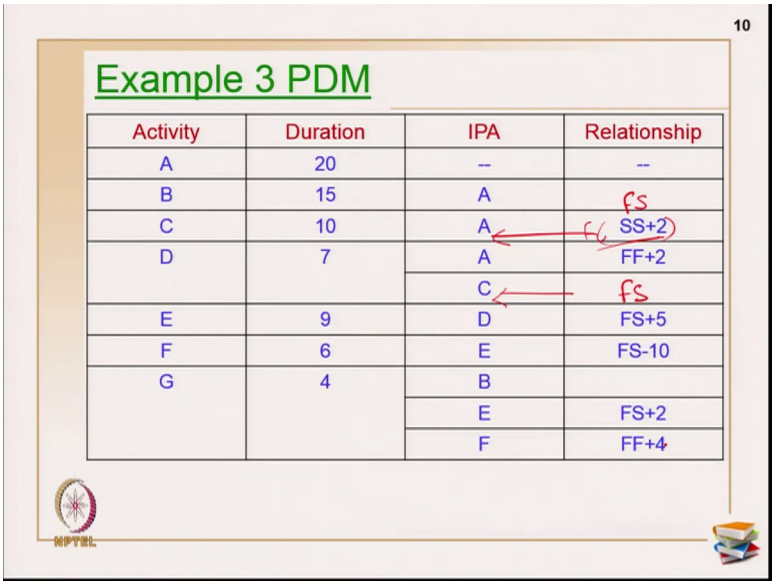
So, I am taking this table which contains activity ID's and the IPA, I have a list of activities here and the list of IPA's which I got from the earlier example are also here. So, first activity A as seen here, there is no IPA and here in this PDM network you need not follow the logic on left to right of arrow marks that you will see here, arrows will come and go like anyway, ok. So, this is activity B, so there is nothing given on the relationship type, if no information is given on the relationship type the default relationship is primarily the finish-start. So, I am using finish-to-start on activity B, then I have activity C, again no special information is given, so it is default finish start. Then I have activity D, then activity E is primarily finish start with C and D, then I have my activity F ok; activity F it is Finish-Finish with activity E then I have, for activity G, I need to have 2 predecessors and J, I have not drawn so far.

So, I am now going down to the line, wherein I does not have an IPA's so I am going to start from I. So, I have finished till F, now I am starting with, so all these are done. So, I am going to start with I again, I does not have an IPA; the next is J, so J has an IPA with E, so this is a finish-start default relationship. Then I have H, H is also with E, finish-finish with H ok, so this is also done. Then now I am going to do activity G because I got the predecessors right now, so it is F and J are my predecessors for activity G; so I have drawn this here. The next I have activity K, so it is primarily activity H and G; then I have L. So, L is Finish-Finish with respect to K, so I have drawn here. Then I have activity M, so it is primarily I, K and L ok; and then I have activity N.

So, after this I have activity O, I have used a cross line arrow marks here, with this is with the predecessor is with E ok, this is also wrong. So, it has to come from till the ok, it should not come in the middle, it has to come to the end of this box, ok. So, this is how I draw my network diagram. So, if you see here, follow some pattern and avoid, as far as possible try to avoid crossing over of arrows, this O could have been drawn here and leaving a space on either sides, this O could have been drawn here, so that this crossing over is not can be avoided ok, that is one suggestion.

And other thing is here you will see lot of dangling activities are there, this is a start dangling, this is a start dangling, this is a start dangling which implies there is a lot of time or float for all these activities. And if you see, this is a finish dangling activity, this is a finished dangling, this is also a finished tangling activity ok; otherwise most of the other activities are all connected. Now, you should be very careful with the dangling activities because they may take lot of floats, ok.

(Refer Slide Time: 26:03)

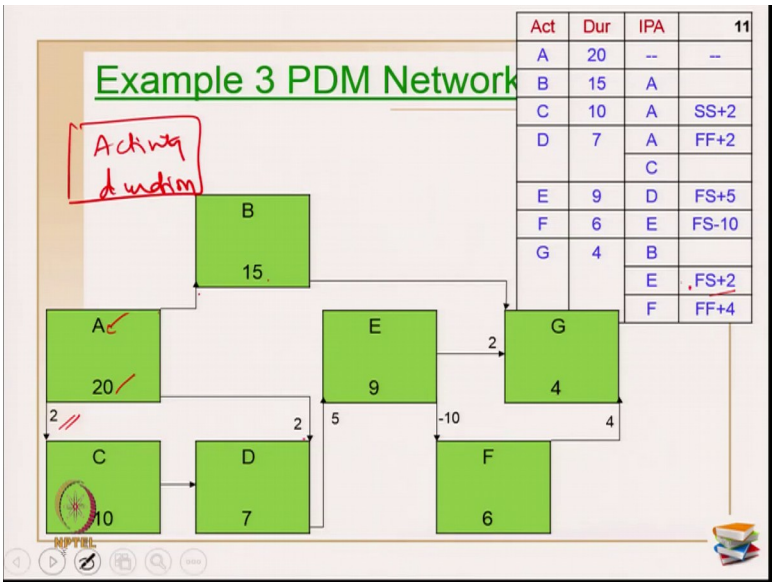


So, this is how the PDM network is drawn, I will show you other examples also. Now let us move on into network analysis, as I have told you while starting with the lecture. There are two ways of network analysis; one is for assuming that the activities execute continuously without any break once started, in the other which is called contiguous activity execution.

In the other case the activity will have an interruption in the middle and it need not continuously be executed till the completion ok. There are two ways, accordingly the calculations on critical path also varies in both the cases ok. Now, this is a simple example I have I am going to use, hypothetical example, so that I can show all the relationships and also I can explain, so I took a hypothetical case.

I have seven activities here, activity A duration 20, no IPA and hence no relationship; activity B 15 days of duration, activity IPA is A; C 10 A SS 2. So, this mean there is; sometimes if I have a different activity here for example, F or something; then there is a special relationship holds good only with the F activity ok; and in this case since nothing is given, so this is special relationship is only for the activity A. When nothing is given for example, like activity C then the default finish-start is always there ok; here also is the Finish- Start that is understood. And this is finish-start plus lag of 5, this is lead 10, this is lag 2, this is lag 4. So, these are the relationship I have used for my hypothetical case.

(Refer Slide Time: 27:41)



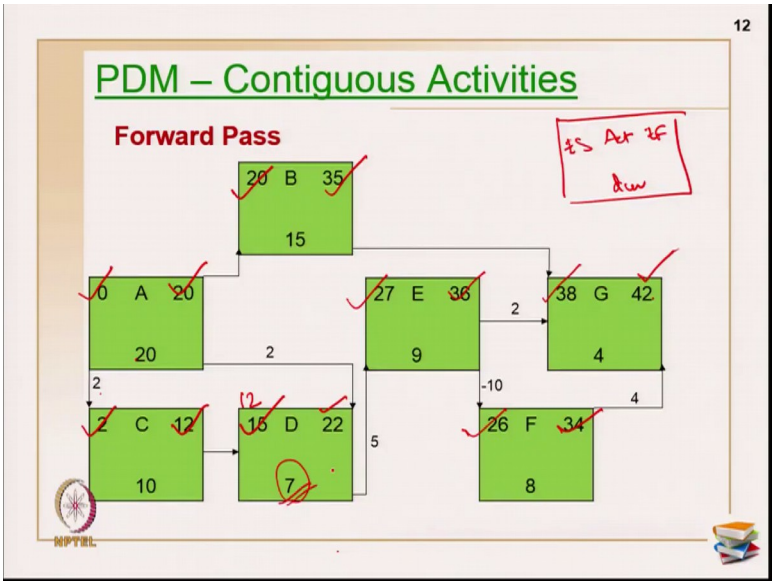
The same example I have shown here so that, it is easy for you to follow how to draw the network. So, I am having activity A which does not have any IPA at all A and duration is 20, so this is a network representation. Along with the critical with the network analysis, I will show you how to make use of the boxes. Right now, I am only using my activity name which is written on the top and the duration is written in the bottom corner ok, that

is how I am representing. So, this is my activity A ok, duration is 20; now next is activity B, I am going to use this is a finish, this is a start relationship and the duration for B is 15 I am keeping it here. The next is activity C this is starts-Start and with the lag of 2 and with activity A and the duration for C is 10 ok. Next activity D I have finish-finish relationship with A with the lag of 2 that is shown here and there is a default C also. So, which implies default finish-start relationship with C is also there ok.

Next is activity E; activity E has a predecessor with A and the relationship is finish-start, but there is a special condition given on 5 days of lag ok. Then I have activity F; activity F is primarily finish-start with E, but a lead of 10 ok; then I have activity G, default finish-start with B, then I have finish-start with E with the lag of 2, finish-finish with F with a lag of 4. So, this is how I have to represent my network. Sometimes, so I may also have an activity name coming in here, then I have to use this combination only for that activity; and if there is something else given here, then that default relationship also has to be taken, ok.

Never try to overdo the relationships, also if no if see whether special type of relationship is given, then that only has to take the dominant role. If no special type of relationship is given, then the default finish-start will be the default relationship for the particular IPA; that you have to keep in mind ok.

(Refer Slide Time: 30:13)



So, now the network is drawn ok, now we will work out for contiguous activity execution, ok. Now contiguous activity execution it is same like your CPM network analysis on AON. Primarily you have to calculate early start, early finish, late start, late finish and early start, early finish for the forward pass, late start, late finish for the backward pass and you will be calculating total float, ok.

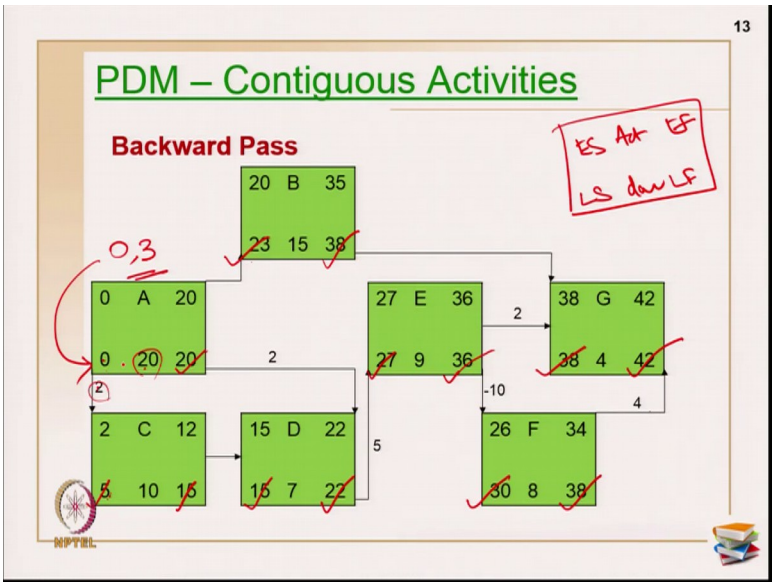
And one thing you should keep in mind is and in contiguous activity execution and activity once starts has to complete till the end and here you may have different types of relationships. So, you have to take care of the different types of relationships and see that the activity is not given a break in the middle ok. So, for example, I will just show this; so, this is my early start and this is my early finish; that is a notation I have used.

So, I am starting with A because I know A was the predecessor ok. So, this is 0, 0 plus 20 is 20 ok so, 0 plus 2 is 2 here, 2 plus 10 is 12 here. I can also do this in parallel. So, this is 20, so 20 is the starting, 20 plus 15 is 35 ok. Now this arrow goes till G, so I have so many activities in the middle. So, I will just go with D now, now D. for activity D this is 12, so it is 12; but what happens is, this takes a duration of this takes this stops at 12, A activity stops at 20, 20 plus 2 is 22 so, this D has to finish by 22.

If D has to finish by 22, if I am going to go with 12 what will happen, I may have a break in the middle because my duration of D is 7. So, what I have to do is retain the 22, 22 minus 7 becomes 15 here; that is a reason of keeping 15 here because activity should not have a break in the middle. If I am putting in this has 12 then D can start have a break or D can have a delayed start or D can finish early and then stop at 20. So, those are all not acceptable in a contiguous execution ok. And now 22 plus 5 is 27 for early start of E, the earliest I can finish E will be 27 plus 9 is 36, and now 36 plus 2 is 38 and this is 35 maximum is 38; so, I am going to go with 38 ok.

Now, 36 minus 10 is 26 use with the signs, so that is easy for you, 26 plus 8 is 34. Now 34 plus 4 is 38, 38 plus 4 is 42 so, the maximum is 42 and we are stopping it at 42. Just go with the CPM network we used, early starts early finish same logic; only thing is because of the type of relationships. You have to see to that we are not allowing a break in the activities, under the same time we are matching up with the activity durations also; that we have to keep in mind.

(Refer Slide Time: 33:17)



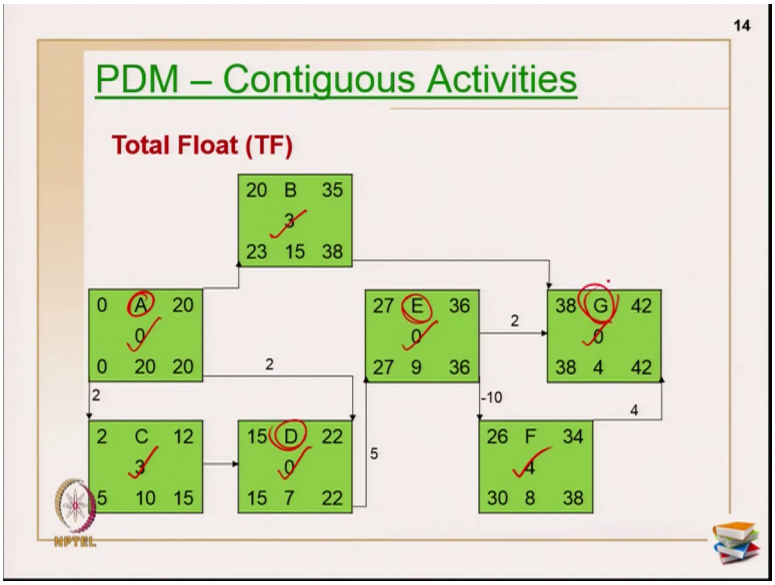
Now, backward pass so same logic here also. So, I am having here activity ok, this is duration, this is early start, early finish from forward pass I have done, late start late finish from a backward pass I am going to do ok. So, I am going to stop, I know that I stopped at G, so I am going to start from here. So, this is now 42, 42 minus 4 is 38, so this is 38; 42 minus 4 is 38, 38 minus 8 is 30 here, ok. So, this is 38 minus 2 this is 36 and this is 30 minus of minus 10 is 40 actually. So, the minimum only we have to take into consideration. So, this become 36 and 36 minus 9 is 27 ok. Now this is 38, 38 minus 15 is 23 so, this is 23; 27 minus 5 is 22 here, this is 22 minus 2 is 20 that also we can hold it up here ok; 22 minus 7 is 15. So, this is tally then 15 minus 10 is 5. So, this till here we have done.

Now, coming back to this, so this is 22 minus 2 is 20, 23 this is 23 minimum is 20. So, I am going to use 20; 20 minus 20 is 0 and this is 5 minus 2 is 3. So, the minimum value is 3 ok. You have to take care of the relationships also and you have to take care of the durations also, which implies here itself I will explain one more time; 20 minus 20 is 0 here ok, as a result of the duration of the activity. So, this becomes 0, with the help of the predecessor relationships this becomes 5 minus 2 which is 3, ok.

The minimum we have to take for the backward pass I am going to use 0 here. So, you have to see for all the relationships and the activity also when we are working on the

backward pass. So, we have completed forward pass, we have also completed the backward pass.

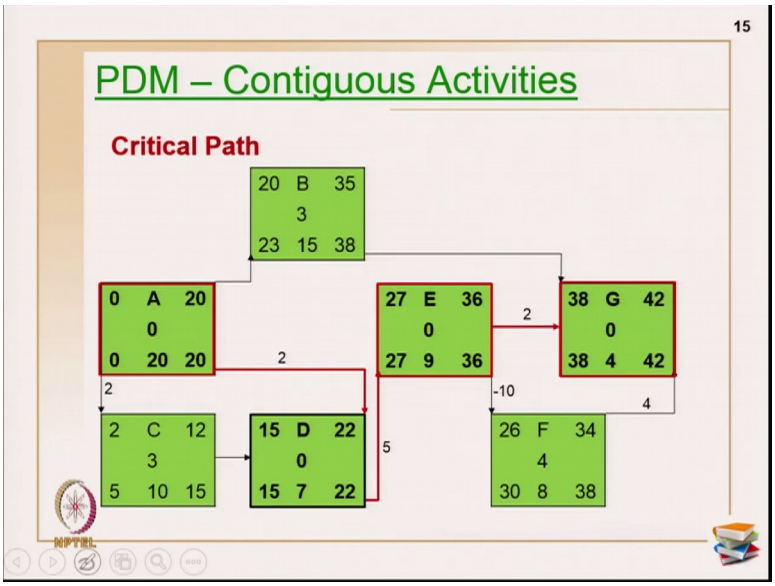
(Refer Slide Time: 35:23)



Now, total float for the all activities; total float is same like a earlier case which is nothing, but you are your late start minus early start or late finish minus early finish so, anything is fine. So, what will happen is some, so late start minus early start is 3 or late finish minus early finish is 3; whatever you want you can try. So, for activity B it is 3, 20 minus 20 is 0, 5 minus 2 is 3, 15 minus 15 is 0, 27 minus 27 is 0, 38 minus 38 right is 0, 30 minus 26 is 4 here ok; this talks about total float of an activity.

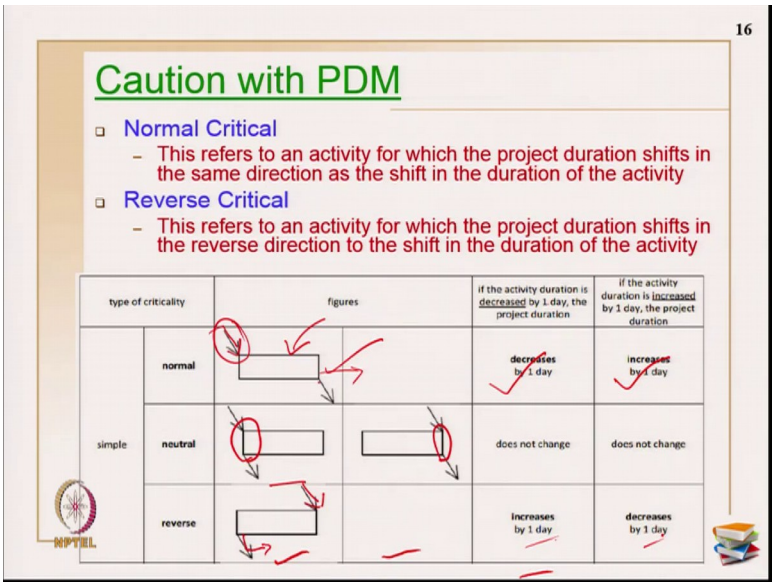
If you see here what are all the activities which have 0 as a total float I have activity A, I have activity D, I have activity E and I have activity G ok.

(Refer Slide Time: 36:15)



So, the critical path is something like this activity A, then I have activity E and I have activity G. So, all these four activities are on the critical path. So, that is a meaning of drawing this contiguous activity execution ok.

(Refer Slide Time: 36:31)



Now, let us move on to caution with the PDM, when you are working with the contiguous activity execution then there are some problems with the finding on the critical activities ok, just with the relationships. This problem will not happen when you are working on interruptible activity executions, ok. Now few things I have brought it up

and like this there are so many cautions are there available ok; I am just going to compare with normal critical, neutral critical and the reverse critical ok.

Normal critical you know what is it, suppose I am having a critical path and one of the activity in the critical path if I am exceeding the duration, the entire critical path will be reduced to one day of the duration ok, project duration will be reduced to one day. Suppose I am taking one activity in the critical path and I am just decreasing it's duration, the entire critical path will have a one day decreasing duration. So, whatever changes you make on the critical activities, the project duration will accordingly you know show it's reflection. If you are reducing it 1 day, project duration also will reduce to 1 day; if you are increasing activity duration 1 day, project duration will also get increased to 1 day, that is what is a normal critical, ok.

Now what, that is what I have written it here and how does a normal critical relationships are in the PDM. Any relationship can end on. So, this activities are called the other normal critical activities. The relationship will have and start it will have some relationships and there may be some relationships which are continuing from the end portion ok, these all will have normal critical. And what; when the activity duration is decreased by 1 day project duration also decreases by 1 day; the activity durations increase by 1 day the project duration increase by 1 day. This you would know in CPM, it works well for all the normal situations ok.

There is something called neutral critical and reverse critical which is what I have written here. Reverse critical this refers to an activity for which the project duration shifts in the reverse direction to the shift in duration of activity. If I am decreasing the activity duration by 1 day, project duration is increasing to 1 day; If I am increasing the duration of activity to 1 day, the project duration gets decreased to 1 day, ok. So, when you are scheduling the network and you are assigning the relationships as I shown in the previous figures and all, you can have combinations of relationships; either I can show finish start plus lag or I can use finish-finish plus lead that is, what is the example I have shown.

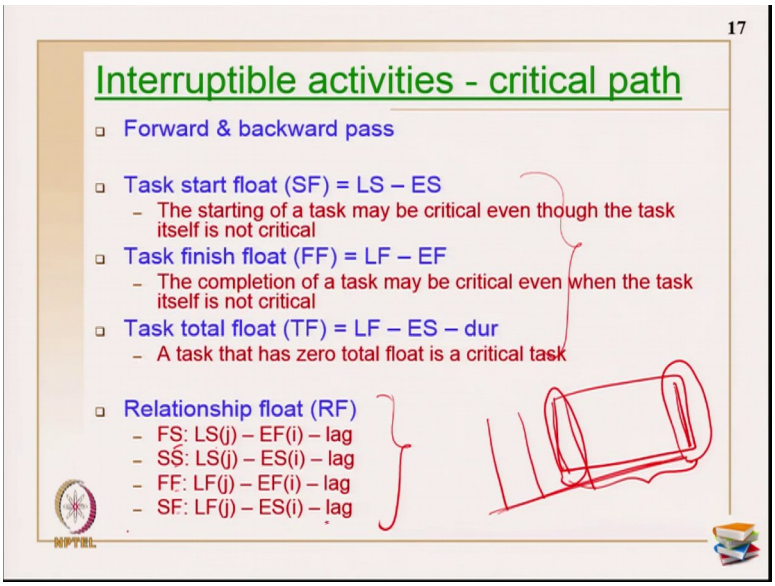
So, when you are trying to map the relationships you should be very careful on or you are landing up in all these cases, ok. That you have to be caution and this analysis on contiguous execution only you have to be careful; in interruptible you would not have

any issues with the same ok. This is a neutral; neutral what happens is all the relationships will end on start or finish and those relationships will also start from the start or finish ok; it is like one sided relationship. Then what will happen in neutral, there is no change at all when you are increasing these activity duration, if you are increasing it to 1 day, there is no change at all to the project duration, ok.

In reverse critical the relationship has to be you know, the predecessor has to be on finished side and the starting the successors relationships has to be on the start side of the activity. Then these you will have a reverse critical type of an relationship, when the activity duration is decreased to 1 day, the complete project duration will be increased to 1 day; when the activity duration is increased by 1 day here, then the complete project duration will be decreased to 1 day, you can try with any example later on ok.

So, you have to be very careful with the contiguous activity execution, only on certain characteristics you have to be very careful. Like this we have bi-critical, multiple critical; there are so many types we have to be really cautious with the PDM analysis, ok.

(Refer Slide Time: 40:51)



The next is interruptible activity execution; this is little tedious type of a calculation. So, some researchers said, when you are using a PDM network and having all the issues on overlaps and so on; then there should be a sufficient number of breaks in between the activities execution, otherwise there is no fun in developing a PDM network.

So, there is another group of researchers who really experimented on interruptible execution and there is a series of formulas to calculate the critical path. There is a slight difference in how do you calculate for interruptible. So, with these formulas it becomes easy for you, I will also explained what is it. There is something called Task floats and there is something called Relationship floats. So, you have to execute task floats also and you have to execute relationship floats also, while you practice on more examples this becomes very easy as well, ok.

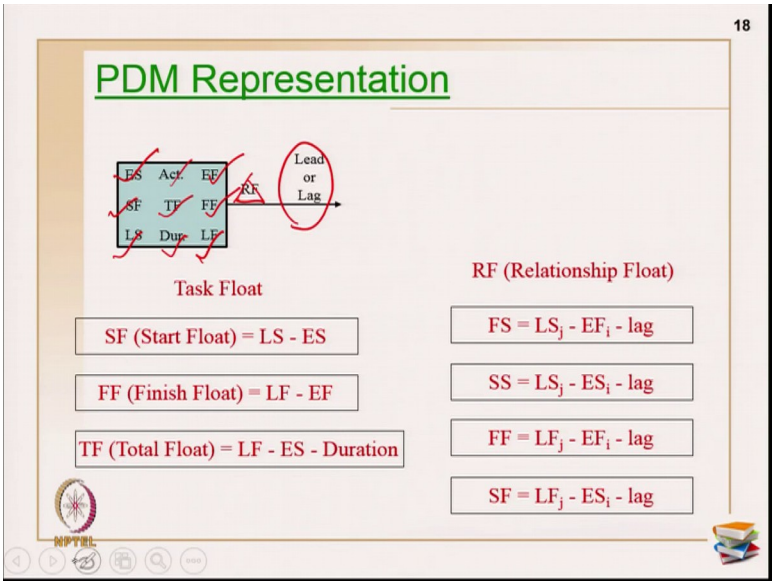
So, I had a task start float. For example, I am having an activity ok, the start of the activity alone is critical and I can start, I can give a break, I can continue or I can stop; the finish or the completion is not critical that is primarily called the task start float ok. In finish float, only the finish of the activity is critical and the start can have the real large float; I can start on this day, I can start on this day, I can start on this day. Only thing is I have whatever is the means, I have to finish the activity on so and so day because the finish part of portion is really critical. In the tasks total float, I have to start the day of the activity on a particular date, I have to also finish it on a particular date and the complete activity is critical. Sometimes there can be a situation wherein, the start float is also critical, the start is also critical, finish is also critical; but the entire activity need not be critical ok, you will see situations later.

So, suppose if we want to have a total activity as critical; then that start-finish, total float everything should be critical in the particular activity ok. So, now, let us see this task start float it is nothing, but formula is late start minus early start. And what is the meaning of the task float? The start of an activity or a task may be critical, even though the task itself need not be critical ok, and task finish float which is given by the formula as late finish minus early finish. The completion of a task may be critical, even then the entire activity is not critical; it need not be critical ok. In task total float, the formula is late finish minus early finish minus duration here then the activity that has a 0 total float is primarily called a critical task. And an activity may also occur where-in the start is critical, finish is critical; but the entire activity need not be critical ok, this is for the activities.

Now since there are many relationships comprising of these activities, we should also calculate where is the relationship floats available, ok. So, for each type of relationships, I have there is a formula given here. For finish start relationship, the formula is late start

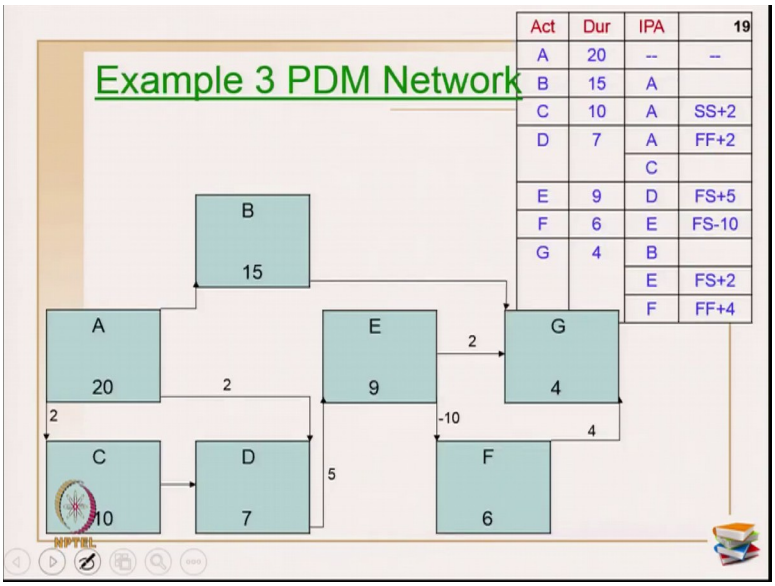
of the successor activity minus early finish of the activity minus lag; for start-start relationship, late start of the successor minus early start of the predecessor minus lag; for finish-finish, late finish of the successor minus the early finish of the predecessor minus lag; for start finish, late finish of the successor minus early finish start of the predecessor minus lag, that is how the formula is comprised in the relationship floats.

(Refer Slide Time: 44:39)



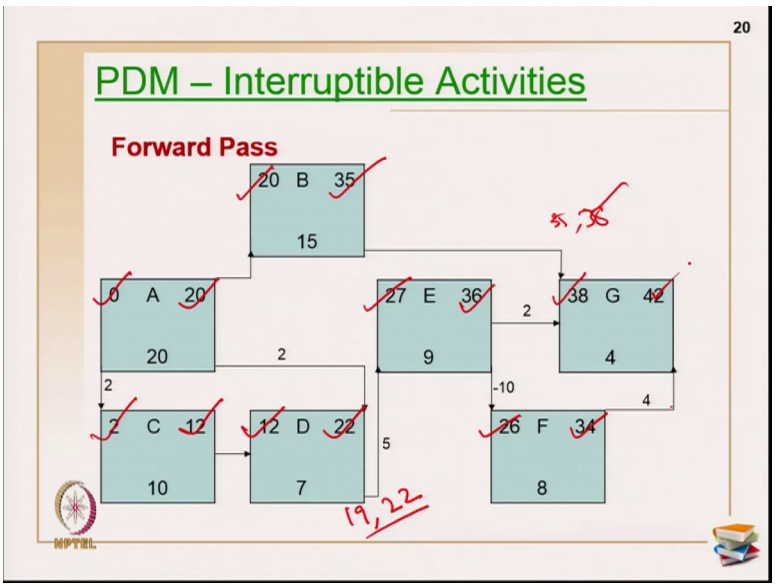
Now, let us start with the representations ok. How do I represent? So, divide the box into 9 zones ok. This is my activity, this is my duration as see earlier, this is early start of the activity, early finish of the activity, this is late start of the activity, this is late finish on the activity; start float is written here, the finish float is written here, total float of the activity is written here. The relationship float is generally given by a triangle, because I can also represent my lead or lag on the same arrows. To avoid no mismatches relationship floats are generally written with the triangle symbol; and this is how the representation goes on and the formulas are also written here the same formulas ok.

(Refer Slide Time: 45:27)



Now, the same example I have taken as explained earlier, the same example; so same network I have right now.

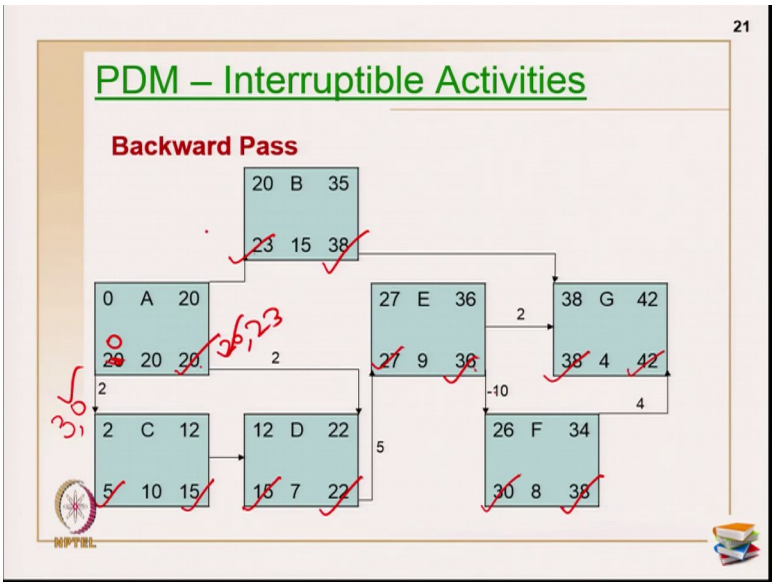
(Refer Slide Time: 45:35)



Now, Forward Pass let us do for Interruptible Activity execution. So, what do you do in forward pass? So, forward pass primarily calculating the early start and early finish of the activities. So, I am going to start at A; so, this is 0, 0 plus 2 is 2, 2 plus 10 is 12 ok, 0 plus 20 is 20; and now this is there is no lead lag or something, no other arrows are also coming, so, this becomes 20, 20 plus 15 is 35. Now let us start moving on here, this is 12

ok. I can still put that 12 because this is an interruptible activity execution; 12 plus 7 is 19 here ok, and this is 20 plus 2 is 22. Now what happens, the maximum only you have to take; so now, this becomes 22; 22 plus 5 is 27, 27 plus 9 is 36. Now, if you see here 36 plus 2, 38 and this is 35. So, the maximum is 38. So, this becomes 38, 36 minus 10 becomes 26, 26 plus 8 is 34, again 34 plus 4 is 38, 38 plus 4 is 42; maximum is 42 and this becomes 42 now, ok.

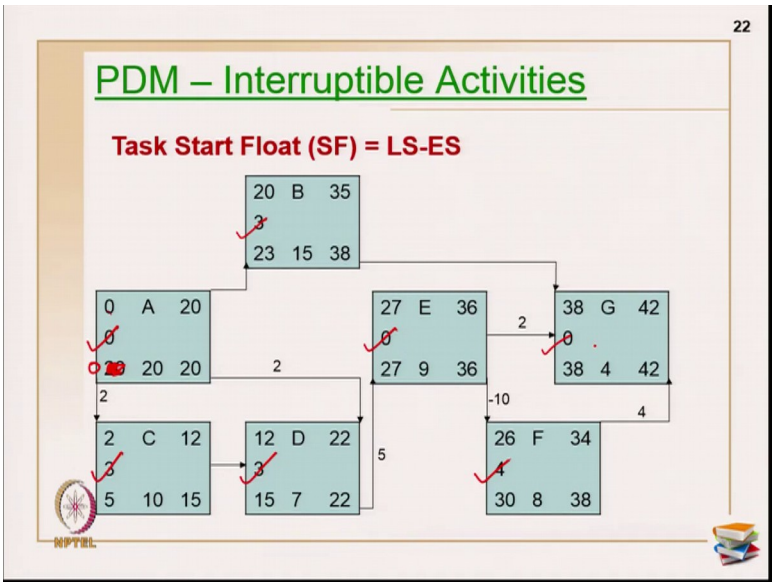
(Refer Slide Time: 46:55)



Let us do the Backward Pass. Now backward pass, I am starting from the primarily calculating the late start and late finish of activities. So, this is 42, 42 minus 4 is 38 here, 38 minus 8 is 30 here, 42 minus 4 is 38, 38 minus 2 is 36 and 30 minus of minus 10 is 40 now. So, which is smaller? So, 36 I am going to keep this as 36, 36 minus 9 is 27 this is here; this is 38, 38 minus 15 is 23, ok. Now 27 minus 5 is 22, 22 minus 2 is 20, 22 minus 7 is 15 this is 15, because I am having a finish start 15 minus 10 is 5 ok. Now 5 minus 2 is sorry 5 minus 2 is 3, 20 minus 20 is 0, minimum is 0. So, this becomes 0, ok.

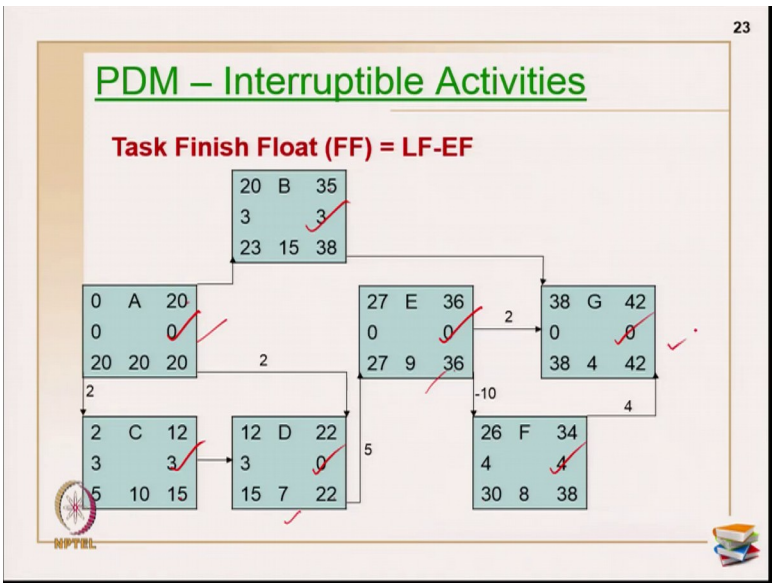
Now, if I see here, now this relationship also I have; so, forgot 23. So, this becomes 23 this is 20, 22 minus 2 is 20, this is 23 only. So, minimum is 20 so, I have kept this as 20 ok; for this late finish of activity A ok. So, I have explained, so, backward pass is also done. Now, just see this is wrong, this is a mistake ok.

(Refer Slide Time: 48:29)



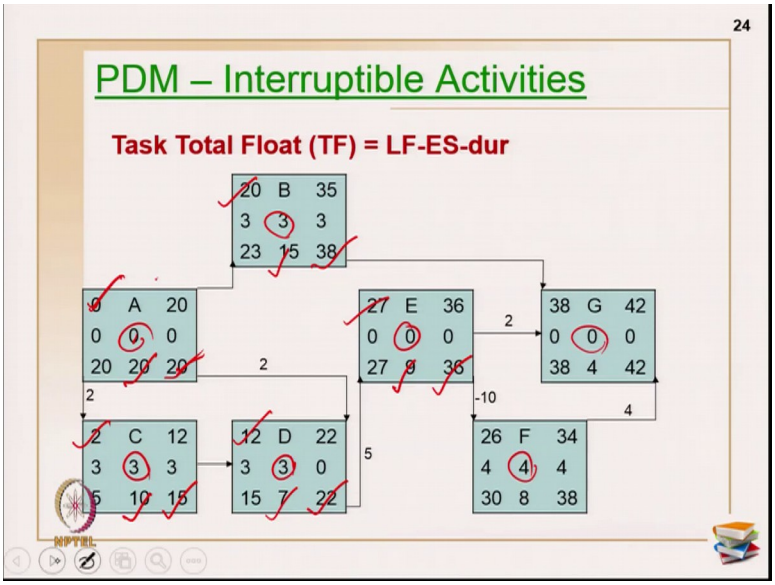
Now we will calculate the total the task floats will start, ok. So, far we have seen only the forward pass, backward pass; now task floats. The first task floats is start float of an activity, primarily late start minus early start; where I am representing the late start, so late start is here, the early start is here. 0 minus 0 is 0, 23 minus 20 is 3, 5 minus 2 is 3, 15 minus 3 is 12, see 15 minus 12 is 3, 27 minus 27 is 0, 38 minus 38 is 0 here, 30 minus 26 is 6. So, task floats are all calculated. So, if you see here start of A is critical, start of E is critical, start of G is critical; because I have a floats of 0 values.

(Refer Slide Time: 49:23)



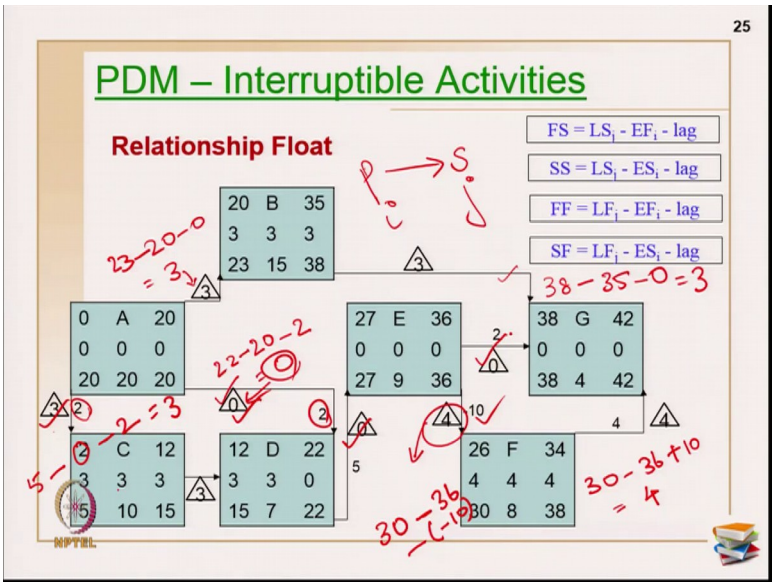
Now, let us do the finish float, a task finish floats, which is given by late finish minus early finish. So, 20 minus 20 is 0, 38 minus 35 is 3, 36 minus 36 is 0, 15 minus 12 is 3, 22 minus 22 is 0, 38 minus 34 is 4, 42 minus 42 is 0. If you see their finish portions; finish of A is critical, finish of E is critical, finish of D is critical and finish of G is critical, ok.

(Refer Slide Time: 49:57)



Now, I am going to see the total task floats ok; task total float, late finish minus early start minus duration. So, 20 minus 0 sorry, 20 minus 0 minus 20, so this becomes 0, 38 minus 20 minus 15, so this becomes 3 ok; 15 minus 2 minus 10, so this becomes 3, 22 minus 12 10 minus 7 this becomes 3. So, 36 minus 29 minus 9 becomes 0, 38 minus 24 minus 8 this becomes 4 and this becomes 0 here ok. So, if you see the total floats in the activities, activity A is critical, activity E is critical activity, G is critical; only three activities completely are critical.

(Refer Slide Time: 50:55)



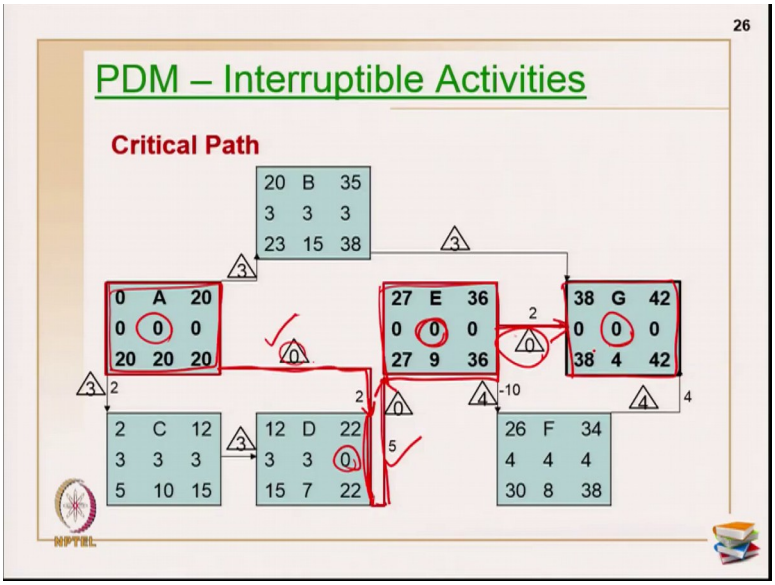
So, how do I represent the floats, I have to still find out my relationship floats ok. Our relationship floats, you have to be very careful and you have to look at the formulas for each of the type of floats ok; and then you may have to work out.

So, this is my same network with all the task floats written here; start float, finish float, total float in the tasks are all written here. And now first let us go with all relationships, this is my finish to start relationship ok. So, this is finish to start. So, late start of this activity is 23, ok. So, minus early finish of this activity is 20, minus the lag; there is no lag value here, which is 0. So, this relationship float becomes 3 here ok. Now again I am going to show with, I am going to show on start-start. So, this is start-start relationship. So, late start of this activity, so primarily you have to take with the successor. So, I am having a relationship. So, this becomes my successor and this becomes my predecessor; this is represented in j form and this is represented in i form ok.

So, my arrow is stopping at here, this becomes my j now again this becomes my i for this particular relationship ok, LS of j is late start of j is 5 minus early start of this activity 0 minus the lead or lag value is given as 2. So, this becomes 3 for me, which is written here ok. Now let us see for finish-finish relationship ok; late finish is given as 22 ok, minus early finish is given as 20, minus that is the lag value of 2 here ok; so this relationship is primarily relationship is 0 here, ok. So, that is how it has to be done.

Now for let us do one more ok, let us take this relationship; this is primarily a finished start relationship. Late start is 38 here ok, minus early finish of this activity is 35 minus there is no lead or lag, so this becomes 3 here, ok. Now, let us do this, this is primarily finished to again start only; but that is a lead value here ok. So, finish-to-start, so this is a late start; late start is 30 ok, minus early finish of this activity is 36 ok, minus use the same minus symbol and then you show ok. So, this is written as 30 minus 36 plus 10. So, how much is this 40 minus 36 is 4, that is how you may have to work out ok. If you see this relationship is having a 0 float, this relationship is having a 0 float, this relationship is having a 0 float. If you have done the calculations perfectly fine you will obviously, see a continuous link going on ok.

(Refer Slide Time: 53:57)



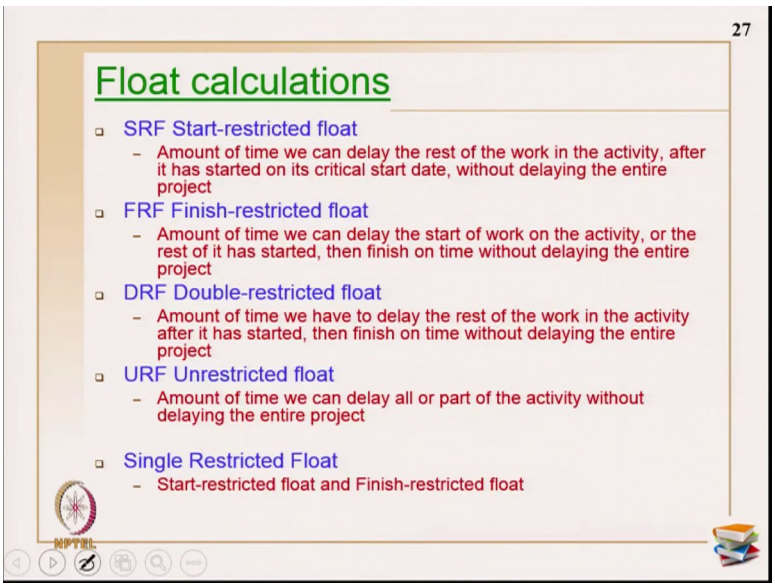
For example there is no break and you will see continuous stretch going on. If you see here, this entire activity is critical because I am having a total task float as 0 ok. And this relationship is critical, because I am having a relationship float as 0; only this finish portion of this activity D is critical, because my this is 0. From here I am having this relationship float also 0, so this relationship is also critical ok. And then I am having my activity E. this entire activity E is critical; because my total task float is 0. Then this relationship float is also 0, which we have worked it out; and now this complete activity is critical because the total task float in G is also critical. If you see here you will follow a continuous path throughout the network and you will easily find the critical path; if you

have done any calculation mistakes you will be easily able to find out in this interruptible execution ok.

So, now so far we have seen two cases in PDM. So, one is contiguous activity execution and the other one is interruptible activity execution. And this is one of the age old method which people use for calculations; finding over the total the task floats, start float, finish float, total float; then relationships, for each type of relationship use a formula and then you will get relationship floats; and then you have to find out the critical path of the whole project.

So, this takes 42 days for completion and you will see everything else is same ok.

(Refer Slide Time: 55:33)



So, now there is an other group of researchers also who have worked out on interruptible activity execution, ok. So, the method of calculation is different, but the answers you will get almost the same ok. So, they have used float formulas only for the activity, there is no relationship or something worked out; and they have defined the floats in a different way ok. So, one is called start restricted float defined as SRF; the next is finished restricted float defined as FRF; the next is double restricted float as DRF; and the next is unrestricted float URF; and then they also have something called single restricted float ok.

So, what is Start-restricted float, you will see the name similar to what we have done earlier, but this method looks really simple compared to this, so I have taught you in this. There is no harm, if you can work out on this method also; there is really no harm in doing it. So, the amount of time we can delay the rest of the work in the activity, after it has started on it's critical start date, without delaying the entire project. So, this looks something like the start critical only ok.

And next is FRF Finish-restricted float, it is the amount of time we can delay the start of work on the activity, or the rest of it has started then finish on time without delaying the entire project. Here for example, if FRF is 0, it implies the activity is finished critical. For example, SRF 0, it implies start of the activity is critical ok. DRF Double-restricted float, amount of time we have to delay the rest of the work in the activity after it has started, then finish on time without delaying the entire project. Here only the start is critical, finish is critical and the activity as such is not critical, ok.

The next is URF, Unrestricted float, amount of time we can delay all or part of the activity without delaying the entire project, ok. This is primarily the float complete float on the activities you can really get with the help of URF.

The next is Single Restricted float, you should have a Start-restricted float and Finish-restricted float both are together, you can call it as a single restricted float, ok. The way calculations are done is little different from how you do; but otherwise you get the similar results as I told you earlier ok, that is all on PDM network.

So, if you look at from the beginning, we have discussed on how to represent a PDM network diagram ok. So, primarily on notations, we have also seen the purpose of building up a PDM network and then from moving on from there we have also discussed the two broad ways of critical path calculations; one is on contiguous activity execution, the other one is on interruptible activity execution, ok. And PDM generally uses AoN only, you cannot use AoA and so in the all the examples also you will see the same thing that is happening. I also explained you how to define the logic, I also cautioned you as to what are all the issues may come up, when you are putting the relationships.

So, when you are putting the relationships see to that, you are not landing up in something called neutral critical or reverse critical or something. So, be careful in how you are representing the relationships, and then you should be able to build up a proper

network ok. As far as software is concerned CPM or PDM, MS project (Refer Slide Time: 59:07) anything any software can handle all the four relationships, along with the lead or lag constraints ok. So, thank you, that is all on PDM, we will see with another new topic in the next class, bye.