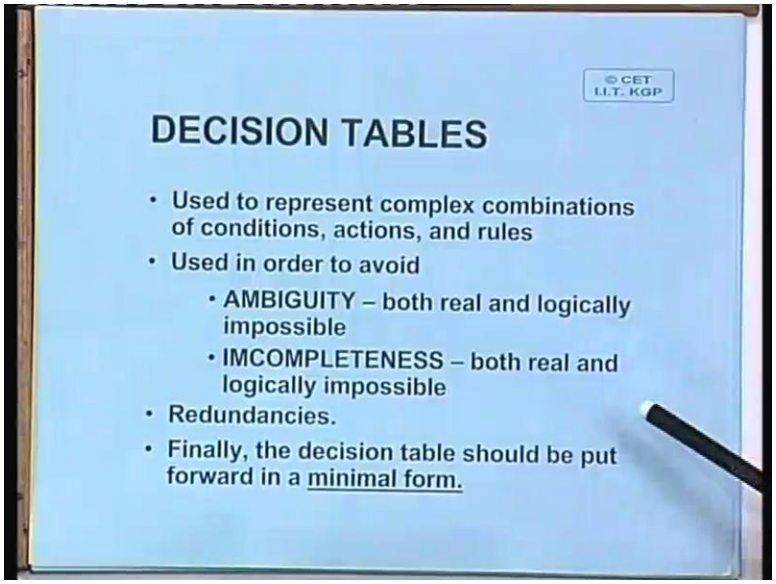
Management Information System
Prof. B. Mahanty
Department of Industrial Engineering & Management
Indian Institute of Technology, Kharagpur

Lecture No. # 14
Decision Analysis II

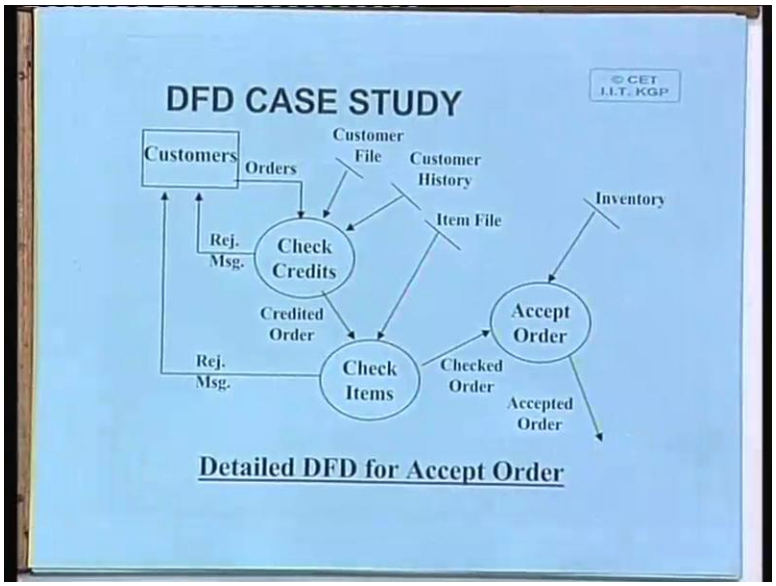
So, today let us continue our discussions with the decision analysis models. Now, particularly today let us cover the decision tables. See, decision tables I have already covered to some detail and let me tell you just repeat quickly used to represent complex combinations of conditions, actions and rules, used in order to avoid ambiguity both real and logically impossible, imcompleteness both real and logically impossible, redundancies and finally the decision table should be put forward in a minimal form.

(Refer Slide Time: 00:01:23 min)



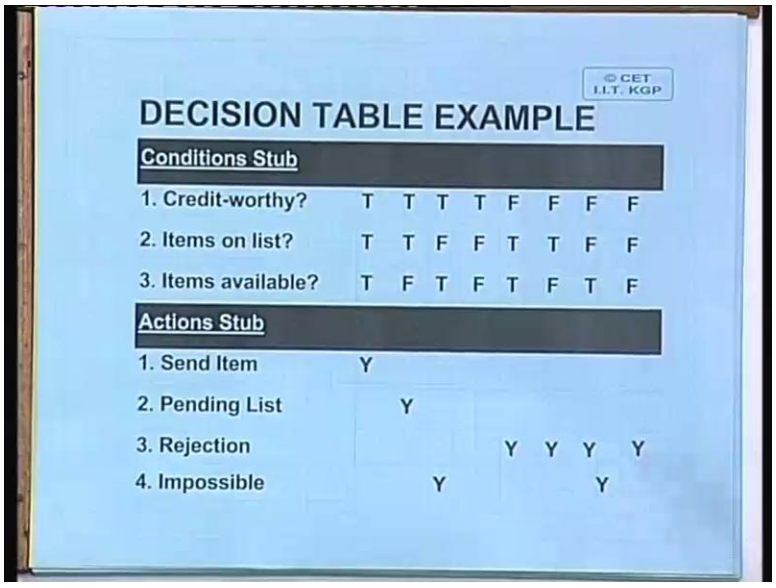
So, essentially there are three steps let me tell you once again. First is the rules, complex rules in terms of logic and all that then try to put them into a logical form by in terms of elementary rules, remove ambiguity and imcompleteness and finally put it in a minimal form for easy implementation, right.

(Refer Slide Time: 00:02:17 min)



So, we take further examples but before that let's go back to that DFD case study. This is the detailed data flow diagram which we have already developed that is customer to check credit, to check item and to accept order and let us see the logic which is put forward in this three particular DFD processes, how it is put forward in a decision table example.

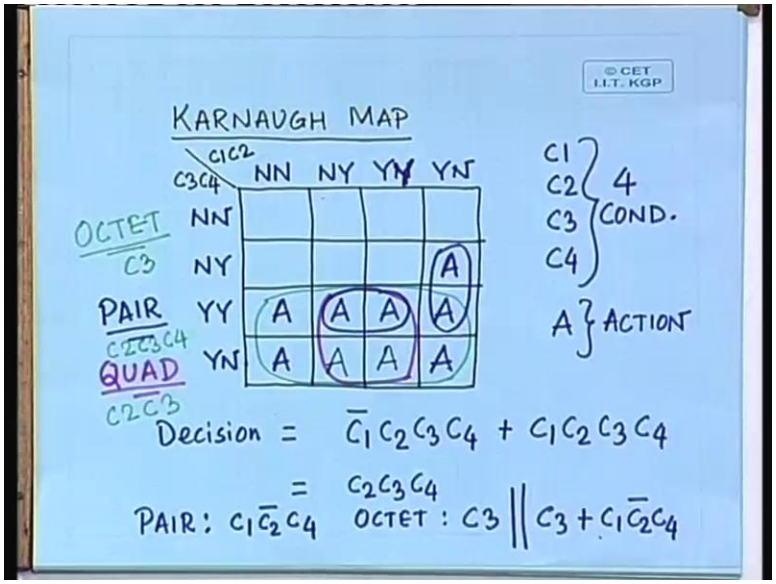
(Refer Slide Time: 00:02:37 min)



You can see here that credit worthy true, items on list true, items available true. So, when all are true we send item, right. Similarly, when first two are true and items available not true then it goes to the pending list. Then in certain conditions basically where it is not credit worthy or where items are not on the list right for those conditions we have it should be go for a rejection slip. But two conditions are there basically your, actually this Y has not come correctly. Actually what has happened that credit worthy true and items on list falls and items available true. This actually is an impossible condition because the item is not on the list but how can it be available, there is some sort of impossibility.

So, there is some ambiguity under those conditions because once it is credit worthy but items are not on list. So, it should show a rejection message but unfortunately what has happened, the item is available. And you shouldn't have checked item availability at all right but you have checked and you have found the items are available. This is something peculiar, so this is an impossible condition. Similarly, the FFT that is credit worthy, not credit worthy, item is not on the list but again item is available. So, there are two impossible conditions in this particular decision table example but otherwise all the things as specified and these impossibilities one should basically take care. What it should take care, that the items on list and items available this table should be matching all right, these tables should be matching.

(Refer Slide Time: 00:05:15 min)



Now, let us introduce another new concept that is known as k map or so called Karnaugh map. See, what happens, what happens that we are having the rules. Now, out of these rules what we need to do, we need to minimize or put the decision table in the minimal form, all right. Now, it you can see that if you put it in the form of a decision table, sometimes it could be little difficult, it could be little difficult to put it in the, to find out the minimum rule. We have said that the don't care conditions and those sort of things, sometimes it can be no complicated to put it in that format.

So, what really is done in a K map is something like this. Let me put you see N and Y basically nothing but true and false. So, you can put them as true false as well. See, you can see that let us say there are four conditions C 1, C 2, C 3, C 4 right and these four conditions on the basis of this four conditions C 1, C 2, C 3, C 4, now these are our four conditions and the action is A and we have an action called A. Now, let us say that what we have done, see otherwise if there are four conditions and each is a binary then how many do you require? There is 16 you know you require as many as 16 boxes. So, look at this, we have 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 boxes here all right and there is a particular way in this Karnaugh map that we have to write.

We have to put NN, NY, YY and YN. Some of you who are from electronics, computer science background you already know this but some others this concept may be slightly knew. Let us say that our action is actually in these two. So, what are the rest? Rest are actually no fine. So, you basically act only when these two boxes are there. Is it okay? How have you got it? Basically, the complex logic that has been given, you have applied the complex logic and you have got that when C 1 is false, C 2 is right true, C 3 and C 4 both are true then you act or in other words you can write the action or decision equal to this is how it can be written. C 1 bar means C 1 is no, C 2 C 2 is yes, C 3 C 3 is yes, look at this A, C 1 is no, C 2 is yes, C 3 is yes, C 4 is yes. So, we wrote it as C 1 bar means C 1 is no, C 2 C 3 C 4 plus C 1 C 2 C 3 C 4. Is it okay? This much is clear to everybody.

Now, what we do usually in K map? You see we have got what is known as a pair. You see if we get two adjacent boxes filled up vertically or horizontally, we can say that we have got a pair, right. So, we have got here what is known as a pair. Now what you must remember if you get a

pair you can remove one variable, all right. So, if you get pair if you can get a pair, you can remove one variable. So, which variable should you remove? C 1, why C 1? Because between these two, look at this. C 1 is both no and yes please look at this carefully between these two boxes C 1 is no as well as C 1 is yes.

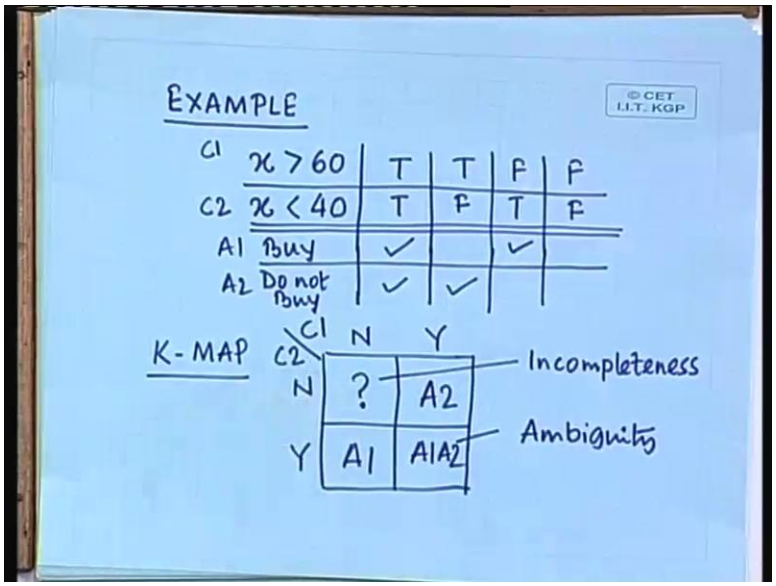
So, C 1 is a variable that can be removed from these two and therefore we can write the decision as C 2, C 3, C 4. Is it all right? So, this is the basic idea about K map but there are more to it. Let us say we have more A's now. So, we have what is known now is a quad. So, let me take another pen. See, there are 4 A's, see quad may not be formed like this, a quad can also be formed horizontally or vertically. So, we can have an entire row as A's or entire column as A's or this way also. So, this is the second thing what is known as a quad. Whenever a quad is formed, you can actually remove two variables. I have said if you get a pair you can remove one variable, if you get a quad you can remove two variables. So, what are the two variables that you should remove? C 1 and C 4. C 1 and C 4, why C 1, because C 1 is no and yes in both this and C 4 because again C 4 is no and yes in this particular thing. So, therefore the quad is actually what is the value of this quad C 2, C 3 all right C 2, C 3.

Then let us go even further let us say we have 4 more A's and all these 4 A's together form what is known as an octet, right. So, we have 8 A's and we have got what is known as it has formed an octet all right. See, it is not that getting any 8 will be enough, they should look like a square all right so, or rectangular shape. So, you see these 8 together form an octet and how many variables you can remove now? 3, you can remove 3 variables. What are those 3 variables you can remove? C 1, C 2, C 4. So, what remains? C 3. So, C 3 will remain, so the value of these octet is therefore C 3. What is the value of the quad? C 2, C 3 and the pair we have already seen C 2, C 3, C 4 okay. So, you can see that this is how the Karnaugh map analysis is to be done. We have to basically look for pairs, quads and octets. If we can get a pair, we can remove one variable, if I can get a quad we can remove two variables, if I can get an octet we can remove three variables. But let us make it little more complicated. Let us say along with this we have an additional A here then how do I put it?

Yes, see actually what we can do, we can make use of overlapping, right. We can make use of overlapping. So, in this case we have got an octet and we also got A pair. So, we can have an octet as well as A pair. So, in this particular case can you tell me what is the value? Why is so complicated? First you tell me the value of the pair. C 1 C 2 bar c 4. Yes, pair is, pair is C 1, C 2 bar, C 4 and octet, so the resulting decision is C 3 plus C 1 C 2 bar C 4. So, you see essentially out of a number of combinations, out of a number of detailed things, we have got all these A's. Then what we have done? We have put it in a simple form that is C 3 plus C 1 C 2 bar C 4. That means in terms of only two decision rules, so C 3 can be put as how C 3 can be put. Let us see that but will see it as an example rather than individual details. C 3 can be shown as don't care at C 1, C 2, C 4 and C 3 at true.

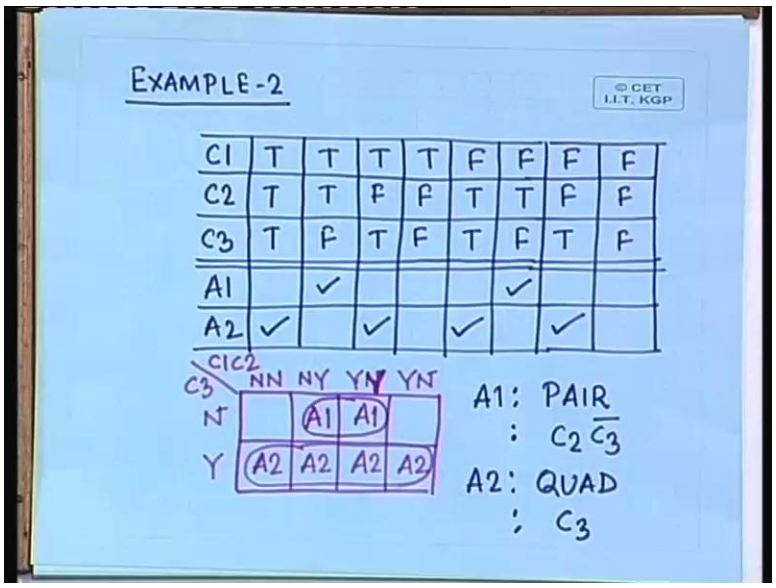
Similarly, C 1, C 2, C 4 can be shown as don't care at C 3 and C 1 is true, C 2 is false, C 4 is true. So, that will be basically our minimal form. In fact if you go one step further, if you can put it in the form of AND gates and OR gates and NAND gates and NOR gates, you can actually design a logic circuit for this. Is it not? You can actually design a logic circuits in terms of AND gate, OR gate and all these. So, suppose this is something about a temperature controller, temperature controller of a thermostat or it is a controller of an electrical machine then you can actually program a logic gate and put that logic gate as an part and parcel of the controller process. This is what basically one should do in engineering but since ours is more of application software kind of a situation, we assume that it will be implemented in software rather than in the hardware, right. So, this is about the basic idea of Karnaugh map. Now, let us take one or two simple examples. Let's go back to our original example, simple, very simple in fact.

(Refer Slide Time: 00:19:27 min)



So, see what we have done here when x is less than 40 then you buy and when x is greater than 60, do not buy. So, this is what we got in terms of K map. When both are true, we get both decisions. When only C 1 is true then we have A 2. See, this is the K map, so this K map essentially shows two things. What are those two things? First one this shows ambiguity and this shows incompleteness, right. So, this is the K map for the simple example and the K map is I mean it is also seen in the original decision table but K map highlights it in a very nice manner.

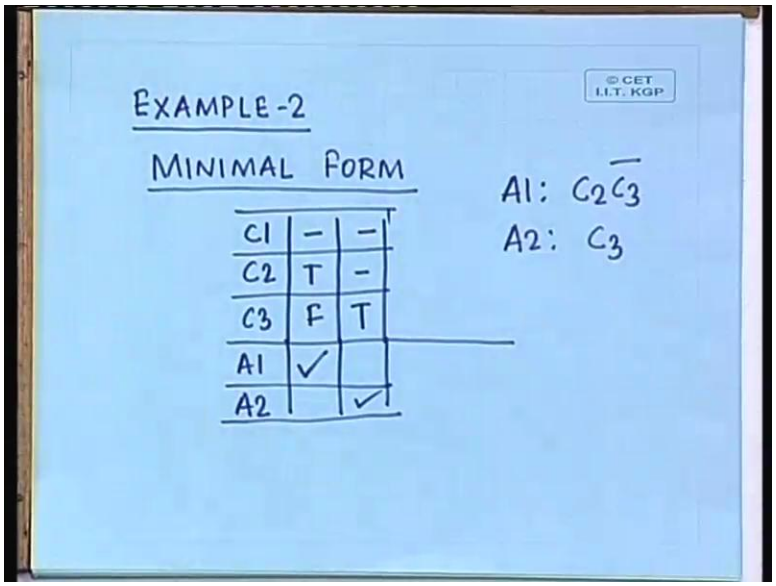
(Refer Slide Time: 00:22:42 min)



Now, let's take another example that is something about and in this example these are the A 1 is why you have TTF and you have here 1 2 3, 1 2 3 yes, FTF. And similarly A 2 is this one's, let's take this particular example all right. So, this one is we have three conditions and there are two actions all right and we have not specified action for 1 or 2 cases. So, you can see that there are some incompleteness but more than incompleteness what we are more interested at this stage is basically finding out the minimal form. So, what should be the minimal form? Try, try your K maps, try K map and from the K map try to find out the minimal form. What you can do, because there are three variables, you can take C 1, C 2 in one direction and C 3 on another direction.

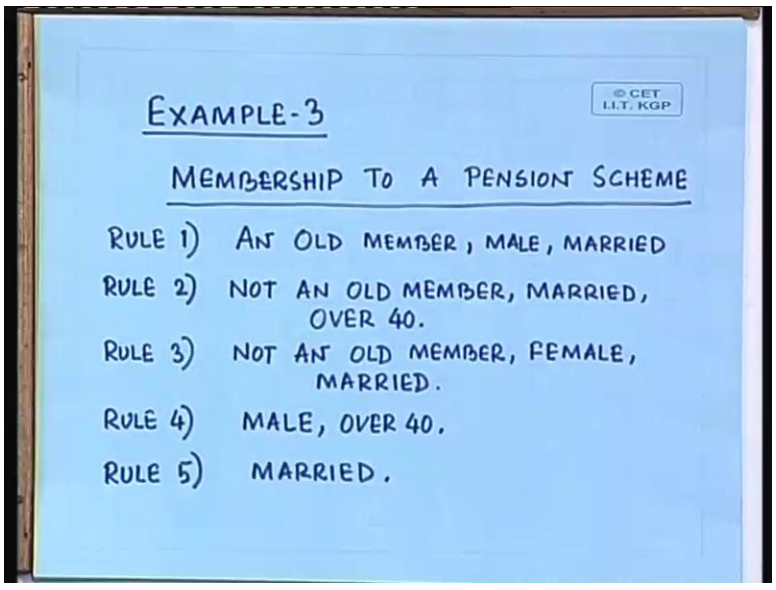
So, you can take it like this NN, NY, YY, YN and NY, C 1, C 2, C 3. So, what kind you have got, which boxes have filled? You can write A 1, A 2 on this. No, first you tell me out of this which boxes are filled up? Okay, so this one. No, this one. A 1, A 1 next one, A 1, A 1 last one, nil, this you know below all are A 2's. You can see that now how do we fill this? By looking at how we have got this decision table from the available thing whatever is given to us, the complex logic that has been given to us. Then on the basis this is what we have found out. So, when we have found out this, we can find that A 1's form a pair and A 2's from a quad, right. See A 1's form a pair and A 2's form a quad, so, from this we can infer since it's forms a pair, it can be put in which form? C 3 bar N N C 2. C 2 because C 1 is both no and yes and C 2 is yes in both the cases, so it is C 2 C 3 bar fine. Similarly, A 2 forms a quad so you can see this is nothing but C 3, C 3 fine. So, this is what we get and using this therefore it should be possible to put it under a minimal form.

(Refer Slide Time: 00:29:35 min)



Let us see what should be the minimal form. So, you see the action for A 1, we have got as C 2 C 3 bar and action for A 2 we have got as C 3. This is what we got from our analysis. Now, therefore we can write C 2 C 3 bar, so C 2 is true, C 3 is false and this is don't care and what we get action as A 1 and A 2 C 3 is true these are don't care. So, this is, this is the minimal form. Is it okay? So, we started with some kind of a complex logic and from this complex logic, we have try to put it in the minimal form, right.

(Refer Slide Time: 00:31:55 min)



So, this is the thing that we have the your K map here and this K map analysis shows a pair therefore C 2 C 3 bar A quad C 3 and making use of them we put it together in the minimal form, right. So, I think this two examples are made things little clearer but still let us take a real example now so that things are even more clear. See, essentially this is about you know a particular pension scheme, this is about a particular pension scheme. Originally the pension scheme was thought of for a group of people right, for a group of people but later on what has happened as time passed then increasingly other angle was also important. Say for example initially the membership, see this is what see the male and female basically the word comes at that time everybody was male, in this company no woman employee was there right.

So, essentially male that is how it was considered and he should be married that is because this particular pension scheme was supposed to be for people who are, who really require it right that was thought at that time due to whatever reason. And an old member because there was an old pension scheme, anybody who is an old member of the old scheme should be taken provided he is male and married right. Then afterwards it was found that many people who did not become old member at that time and those people are nearing their retirement age and you know so increasingly people who are thinking that they should be given chance. So, new condition has brought into that is basically age. So, the new rule not an old member, not an old member married and over 40, right. Then some in this time then they thought that you know this particular rules they do not consider anything about females, right.

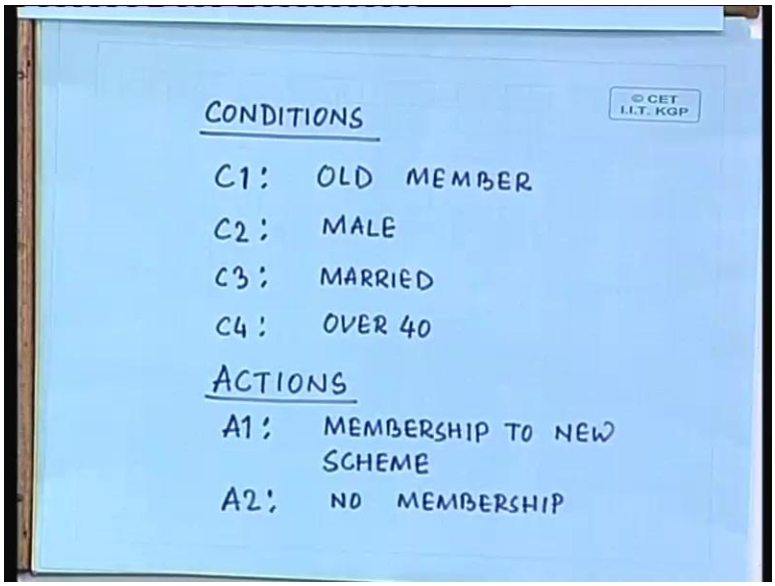
So, let us bring that component also and for that since there was no old member who is female, so it was automatically thought that it is a not an old member, all right, not an old member but this married condition is put anyway right so that was the third rule. So, this is the third rule, not an old member female and married right. Then after this rule some people particularly the old employees they said that there are some people who have not got married right but they are quite old, why should not they get this benefit, right. So, we should give them benefit also however since the females are you know or women employees are newly joined, there is hardly any need to include them. So, let this rule be specifically for those category of people, male over 40 all right whether member or not let us not bother. So, this becomes a rule four, all right male over 40. Now when all these criteria has been taken care of then some people pointed out that a

person who is not an old member we are given some benefits whereas the old members are not given benefit. For example if someone is old member married and over 40, what will happen right. So, you see those kind of anomalies then peoples are okay, forget all this let us make a swiping rule. What is this swiping rule? Anybody who is married will give this facility, all right.

So, finally fifth rule has been made that just married anybody if he is married that's the thing that he should be member of this scheme. So, you see basically in this particulars example what I wanted to point out is that due to whatever reason, due to whatever reason as time changes new concepts come, new thinking comes, new people join, new management comes, the rules are changing. But usually what happens rules as such do not change, new rules are added. Sometimes one can scrap also, suppose one can scrap rule one, possible but usually this is how things keep on happening and instead of elementary rules these rules usually become complex rules which are combination of lot of elementary conditionalities. Now, if you have to put it in a computer form, in the form of a program or process logic then it is necessary that you do a decision analysis, all right.

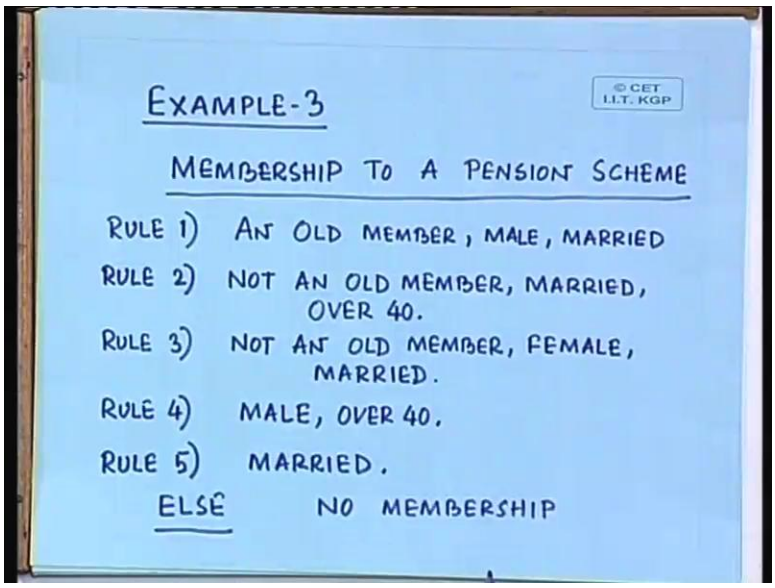
So, how should we begin? We should begin by, how should we begin? We should first find out the conditions and actions right. So, what are the conditions? There are 4, 4 conditions. Yes, 4 conditions let us write down the conditions.

(Refer Slide Time: 00:40:28 min)



So, this is what is the condition and what are the actions, fine. So, we have decided the membership to new scheme as well as no membership, fine. Now, see you can always draw the decision table to begin with. To draw the decision table, how many boxes should be there? There should be 16 because there are 4 conditions and each can have two options true or false. So, we can have 16 combinations and we can specify the actions accordingly but okay basically not necessary but what happens. See, basically we have to see for incompleteness or ambiguity right.

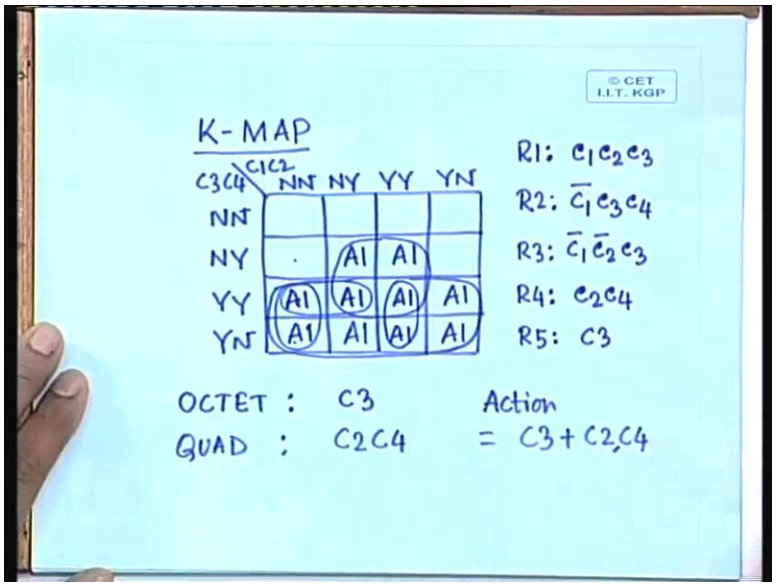
(Refer Slide Time: 00:43:32 min)



So, one action should be enough membership to new scheme but no membership we have also kept because suppose it is said that if not deny. See, look at these 5 conditions, what we have said? Membership to a pension scheme, rule 1 to rule 5 but we have not said what happens if not. We have not said that. Let us say these now specifically else, so else is also a rule. See, this else need not be said all the time, many of the places the accepted criteria is given, the rejected criteria is not given, sometimes the rejected criteria is given, accepted criteria is not given. This is also done deliberately at times let me tell you.

Most of the organizations because they do not know the implications of the no rules, see if you say else no membership these basically means that anybody who is not covered under these may still be eligible. See, nobody has seen so that is how the all these 5 rules have come but if you say else no membership then you have also specified that if you have not fulfilled this, you are out. So, many organizations they do not give an else rule, fine. So, if you give an else rule definitely your decision table is complete because there are some rules else another rule but problem comes only when the else rule is not given. Then you may have some incompleteness situations for which the organization should be given some actions.

(Refer Slide Time: 00:45:38 min)



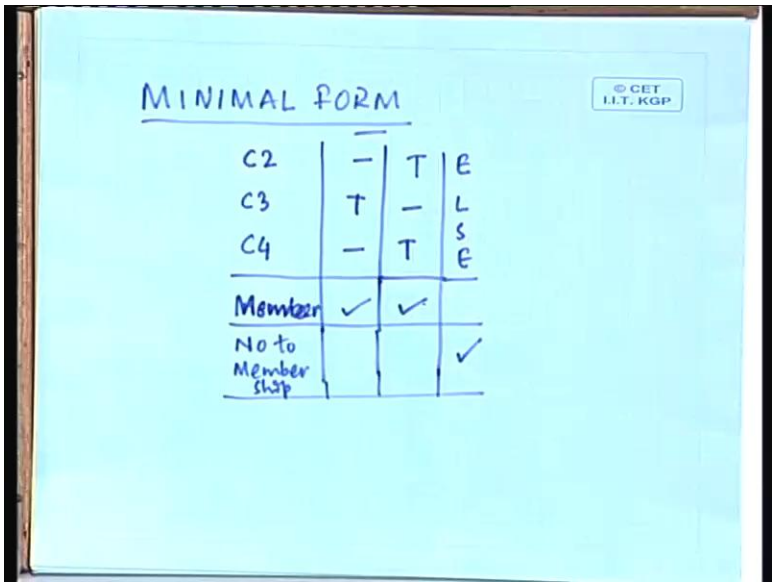
Now, moment you have given an else rule, it is automatically clear that in our decision table we are going to have complete decision table, all right. So, I am not drawing the decision table because it will be little to big that will be around 16. So, I am just putting the K map, just K map let me draw of this particular situation. In fact you try right so NN NY YY and YN, so this is the thing C 1 is the old member, C 2 is the male, C 3 married C 4 over 40. So, we have to draw this K map and if you really put this rules here then let me put it in here, rule 1 an old member male and married. So, old member is C 1, male is C 2, C 3 this is rule 1. Rule 2: not an old member C 1 bar married that is C 3 over 40 C 4. Rule 3: not an old member C 1 bar female C 2 bar and married that is C 3. Rule 4: male and over 40, so C 2 C 4 and finally R 5 that is C 3, fine. So, this is the thing, now we have to specify the actions on this particular case.

So, first we begin with C 1 C 2 C 3. So, what are the things C 1 C 2 C 3? C 1 Yes, is this, C 2 yes, C 1 C 2 both yes is this and C 3 also yes is these two. Is it okay, fine? So, these two we write A 1 then C 1 bar, C 1 bar means this and C 3 C 4. So, these two is it not fine. C 1 bar means where C 1 is no and C 3 C 4 both are yes, so we write A 1 A 1. Then the third rule where C 1 bar and C 2 bar that means both are no and C 3 is yes, so actually these two, I am sorry these two. Already one A 1 is there, so another A 1 then C 2 C 4, C 2 C 4 is a quad kind of a thing. So, C 2 is yes here, sorry yes here and C 4 is yes here. Is it okay? So, this 4, this is C 2 C 4 and finally C 3 yes will make all these as A 1. Is it okay, fine?

So, you can think about it how we have got this particular logic. You can see that there are lot of overlap, lot of overlaps here so let me just put it once more. C 1 C 2 yes is this where C 3 is also yes, so these two, fine these two. Then C 1 bar and C 3 C 4 is these two then C 1 bar C 2 bar C 1 bar C 2 bar so this, this is it not this. And then C 2 C 4 is this one and C 3 is the whole thing. C 3 is this whole thing. Is it clear? So, you can see that between these decision rules. There are so much of overlap because of this overlap usually we may have difficult situations and because they are actually complex logics. So, what should I get now? we simplify very easy to simplify, we have a octet and we have a pair or you can say it's a quad, I mean up to you, you can put it as a pair also but if you want it that way so you can have an octet. So, the octet is this one. What is the value of the octet? C 3. What is the value of the quad? C 2 C 4. C 2 C 4.

So, you can see that all the other rules actually are become redundant. So, we have the action equal to C 3 plus C 2 C 4, right.

(Refer Slide Time: 00:53:02 min)



So, we can quickly put it in the form of a C 2 C 3 C 4 insure. So, what we get C 3, that is true don't care, don't care and C 2 C 4 is true, don't care, true fine. So, this is the minimal form. So, what has happen in the minimal form that we have been able to remove one condition C 1 that is old member or not, we need not check even C 3 true member, make him a member, C 2 true, C 4 true, C 3 don't care make him a member else no two membership. So, this is the minimal form and we can use these rules only as for the present situation. Is it okay? So, this is about the decision tables, I stop here today and there are some more things about decision analysis which we shall discuss on Thursday. Thursday we have exam, sir. Wednesday.