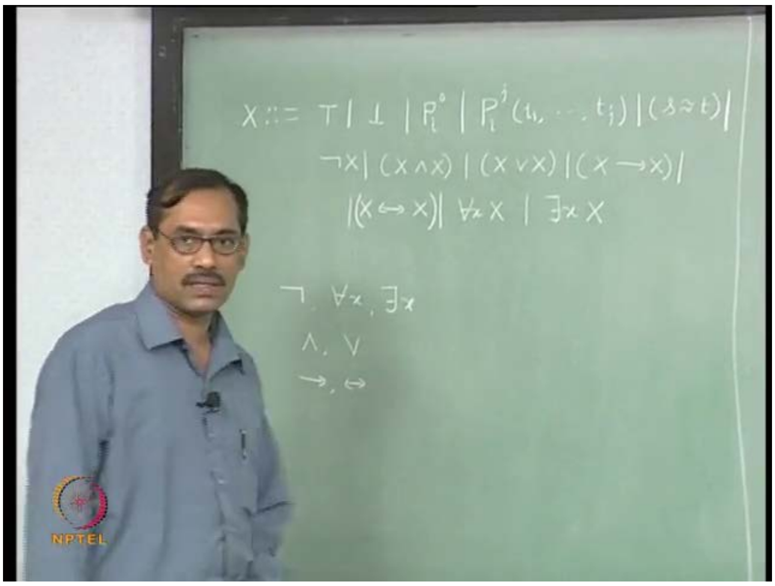
Mathematical Logic
Prof. Arindama Singh
Department of Mathematics
Indian Institute of Technology, Madras

Lecture - 23
Symbolization & Scope of Quantifiers

We had just formulated what a formula was. So one formula we have not introduced there, that was the equality predicate. You can see your notes. So, top, bottom, predicates, everything we accepted, except the equality predicate. The equality predicate is a specific thing. We do not want to give the same format as for other predicates. We will write it in infix notation because it will be easier for us to read; instead of writing equal to a comma b it will be easier for us to say a equal to b. So we will be using the infix notation there. I think the unique parsing will still hold; same kind of proof. We take infix notation instead of the outfix one.

Let us give that grammar again. A formula can be top, or it can be bottom, or it can be any predicates with 0 arity, so there, we are specifying specially, because it will not have any arguments. We write say P i 0 or it can be P i j. Then it should have j number of arguments, to show that, we will not write just t j times will say t 1 to t j instead. We will write t 1 to t j where t 1 to t j are generic terms, any generic terms, they can be same, they can be different, and so on. Then we will include the equality predicate here, say, s equal to t, that is all; in infix notation.

(Refer Slide Time: 01:02)



We are not writing the usual form of this, it would be s comma t, that we are not writing; we are writing in the infix notation only. Then what else we have? We have to take care of the connectives and the quantifiers. We say not x, or x and x, or x or x, or x implies x, or x biconditional x, or there can be quantifiers; there can be 'for each x' or 'there is x x'. Out of these, the first row, they are called atomic propositions or atomic formulas instead, not propositions now. They have no connectives, no quantifiers in those; so they are called the atomic ones. And all the others are to be said as compound ones, compound formulas or compound w f fs, well formed formulas. This is just a terminology which might be useful later.

Now, let us see how to symbolize for usual contexts and how many brackets or parentheses you can omit as per the propositional logic. In propositional logic we had put down some conventions so that some parentheses you can omit while in writing, and it might improve also eligibility. Like the outer parenthesis, you can omit that is only required for the unique parsing. Once you get matured you can forget it and still read it uniquely, fine. That is our first rule. That, we will forget the outer parentheses, we will not write it.

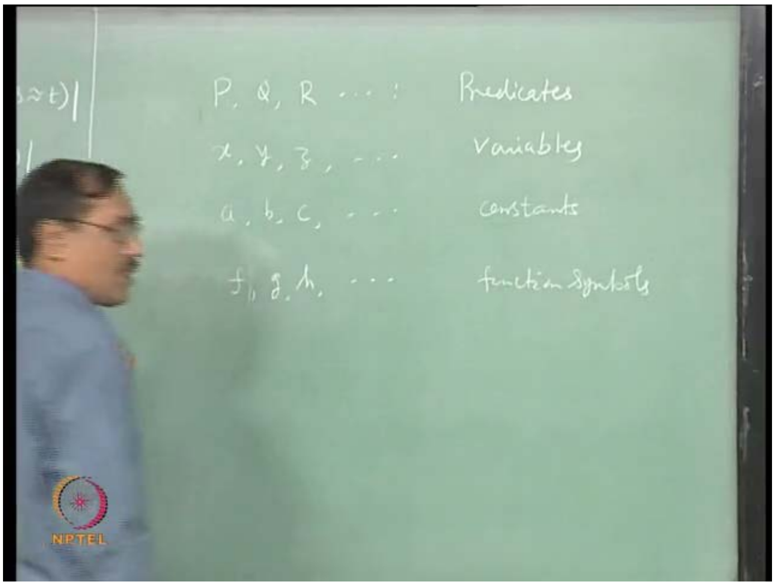
Then some more can be reduced, by laying down the precedence rules. The same precedence rules as per the propositional logic. We will have not as the highest precedence, then and, or will have the next precedence, and implies and biconditional will have the lowest precedence. But now you have the quantifiers also, right. We will put quantifiers at the level of not, right. They look like unary, so we will put them there. We will say not and these quantifiers, for each x x and there is x x will have the highest precedence.

So, our precedence rules will go like: not, for each x or there is x; in fact, all these, for each, there is, always come with some variable. We will generically write for each x there is x. It need not be that particular variable x. These will have the first, or the highest precedence. Next precedence will be and, or, as usual. And the lowest will be biconditional, the implies and the biconditional. This will reduce certain number of parentheses. But not all parentheses are omitted; just like your propositional logic, there can be some which you want to show here. Then it will be easier to read, fine. Sometimes we also use extra parentheses for legibility. We will see that occasion later, where even if it is not allowed, we will put our extra parentheses so that our reading will be easier.

We are not very fussy about parentheses here. But if you have some doubt in reading, then you have to be fussy; otherwise you cannot read it properly or precisely, rather. This is our next convention and some more conventions will put to reduce the cluster in writing the subscript and the superscripts. Suppose in a context you use the, what you use, the predicate P1 and then you have 3 arguments there. Throughout that context you are using 3 arguments with P1. Then it is not necessary to write P 1 3. We will forget that, but we have to be careful there, because sometimes we will write p 1 with 3 arguments and again p 1 with 2 arguments, then there will be problem.

So, we can do that; omit the superscripts, provided you always observe that in the particular context that always you are using the same symbols with same number of arguments. Then, next, what we will do, we will have still subscripts P 1 then P 0 or P 100, and so on then slowly we will forget them also. We will say P, Q, R and will accept as predicates, forgetting the subscripts also.

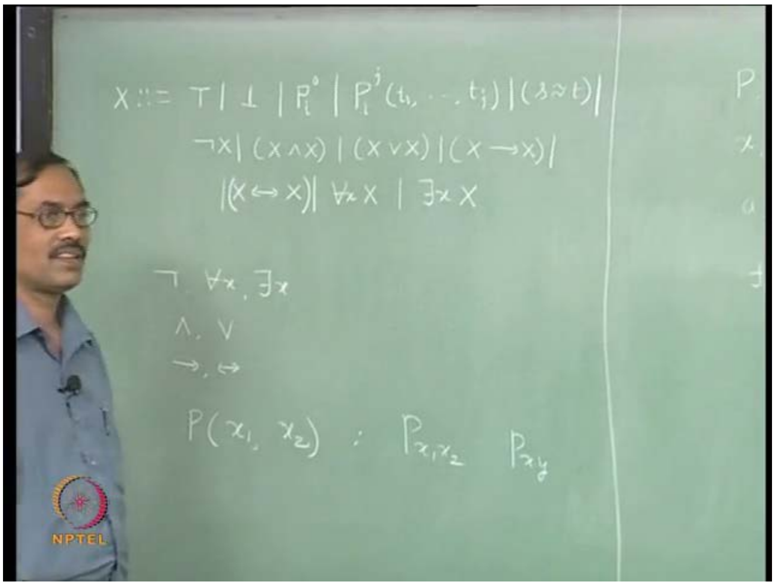
(Refer Slide Time: 06:56)



We can write these things for predicates, but equality predicate will be written as equality predicate. We will not confuse with anything else. Then similarly, for variables we may write x, y, z, and so on, instead of the subscripts x 0, x 1, x 2, instead of all those things, we will use. Suppose in a particular context you have more than say 100 variables then you cannot write with x, y, or z, or any alphabet. So, we will go for again x 1, x 2, x 3 or even x, y, z, or x1, y1, and z1, so on; just to reduce cluster and writing better way.

Similarly, for the constants we will use a, b, c, from the lower end of the small letters. So this should have been written as f i 0, 0 array function symbols. They are the constants; instead of f i 0 that is f 1 0 or f 2 0 or f 3 0, and so on, we will just write a, b, c, small letters. And then again for the function symbols, instead of writing f j i, i may range from 1 to 100, and so on, and j might be number of arguments it takes, we will again say f of that many arguments, always use in the context so forget j; then f i. You can forget by writing f, g, and so on. We will write, f, g, h, and so on for the function symbols. These are variables, constants; these are the individual constants, these are individual variables; then these are function symbols. Then there is one more, which sometimes, you follow sometimes, you do not speak; it is about the commas. In predicates it is customary not to write the commas and the brackets.

(Refer Slide Time: 09:15)



For example, you have say P of x 1, x 2. Instead of these, we just write P x 1 x 2, or even Pxy. But these will be capital letters; they will be small letters. So, they will be taken as arguments. We will forget the commas also. It will sometimes be easier to read, this. Even for function symbols, that can be followed; forget the commas and go on writing. Still you can prove unique parsing, in that without the brackets. But sometimes it may not be easy for us to read. So, we will keep the commas. If it is not easy, if it is easy read, we will write only, one argument is there, just, you can simply write it with or without brackets, so that will be our next convention.

Then now let us go back to our original thing. What we wanted, with all these shorthand writings, that what you get, they are not really formulas. They are sometimes called formulas, abbreviated formulas; formulas which have been abbreviated. We can expand it, bring it to correct form, by suitable vocabulary of which one you are rewriting as what, but we will regard them just as formulas, and go on with that. Yes?

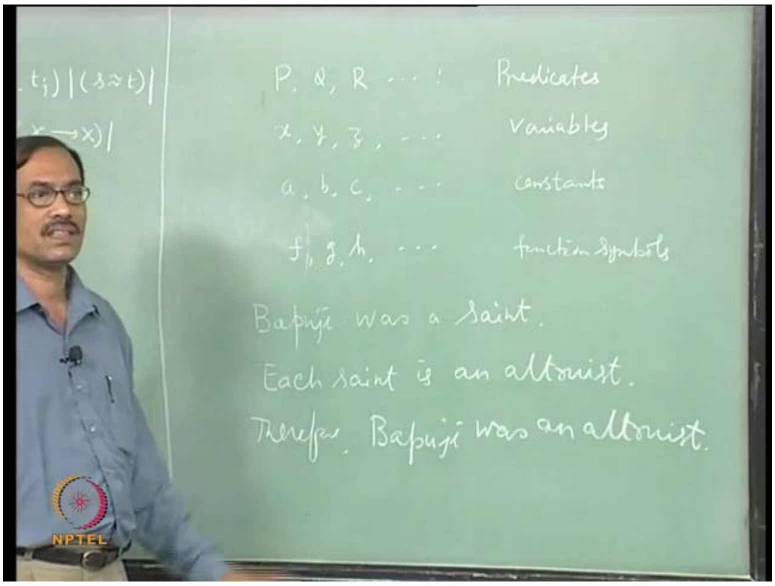
Student: Constants here what does it refer to is that propositional constants or.

Well, what are the constants? The constants come from f with superscript 0; so they are the function symbols having no arguments. And function symbols, we are writing for the definite descriptions, which will refer back to some objects, particular objects from some set, that is our concept. Now, we have not materialized it, right? That means these constants will refer to some persons, particular persons.

Student: What about predicates, 0 argument predicates, they are also constants?

Zero argument predicates; they are not constants. Predicates, when given some values, the arguments are filled in with proper objects or different objects, give you sentences So, predicates without those arguments, that is that superscript is 0, gives you propositions, the atomic propositions. In that sense, first order logic is an extension of the propositional logic, where you have the propositions, you will have the connectives, everything of propositional logic. In addition, you have some variables, function symbols, and the quantifiers.

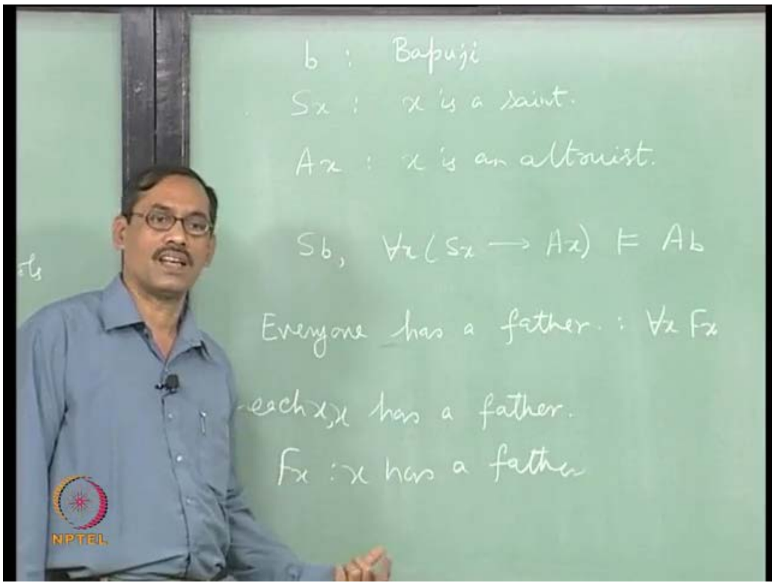
(Refer Slide Time: 12:06)



Student: Sir, give some example of constants.

Will give some example of constants. Let us take our first example. We started with one simple argument. Bapuji was a saint. Each saint is an altruist. Therefore, Bapuji was an altruist. Fine. Let us look at the first sentence. How do we symbolize this? You have to find out, you are going now deeper into the structure of the sentence, not only as propositions. Now, in this sentence, this is one subject and this is the predicate. Grammatically you can say some phrase and so on. But now for us everything will be predicates or names or quantifiers, these are the three things we have to translate with. Here we find that somebody is a saint, that is one predicate. We will build up our vocabulary say, Sx, let us write it first, for x is a saint. Now what about Bapuji? It is a name, particular person, right, this is what we understand from this, in some other context, we may mean something else. This is what we understand, now with our common knowledge. This, we will be taking as a constant. You say a 0-ary function symbol, it might refer to a particular person, a definite description without any arguments.

(Refer Slide Time: 13:15)



So, we will start with b for Bapuji, unlike our English language, they will be small letters. In English language, the initial letters will be capital for the names; for us they are the small letters. Now, the first sentence you can translate into first order logic: it will be just Sb, S with bracket b. But we are forgetting the brackets now, so this sentence can be symbolized. What about the second one? x is a saint, so we need increasing our vocabulary. We do not know how to write altruist. Let us write it first, say Ax, x is an altruist. That is enough for this. This is also okey. Now you can go for the translation. First one is: Bapuji is a saint. We may write Sb. The second is. If x is a saint then x is an altruist, right. So, first you forget the quantifier; you look at it, what does it say? It is not asked for each x, x is a saint and x is an altruist. Usually, it is the rule, when you translate from the English sentences and you have the quantifier as for each, then it will be an implication. it will not be and. It will not be conjunction; usually, that is the rule. Sometimes it may brake down when you come across some mathematical theorems. But usually, when it comes from the natural English, it will come like this.

The second sentence is: for each x, if x is a saint then x is an altruist. Then you have the last sentence; and that will give us the consequence. This consequence is valid or not, we do not know; we are just writing the same way as in propositional logic. We will introduce the symbols later probably. Now, what will it say? Bapuji was an altruist, so A of b, is that okey? They can be symbolized this way.

Let us see some more examples. This will really clarify the meanings, later the meaning in formal semantics we are going to develop. Suppose we take this sentence: everyone has a father. How do you symbolize this, only this much: everyone has a father. Well, you are writing x has a father, x, we do not know what it is, it is a variable.

Student: For all x F x.

For each x, x has a father, not some, right. When you write everyone it does not say for some x, it says for every. So, for each x, x has a father. Now, x has a father, that itself you can take as a predicate. Because in the context we are not going deeper into father or mother or has a. He has a pen, fountain pen, he has a father, this 'has a' can be different, but still you can write as one, write, or it can be different. So, having a pen and having a father may be of different category. There, this same word has been used. You might like to represent it in a different way, depending on the context. So translation is not really a literal translation; you have to look out the meaning slightly and then translate.

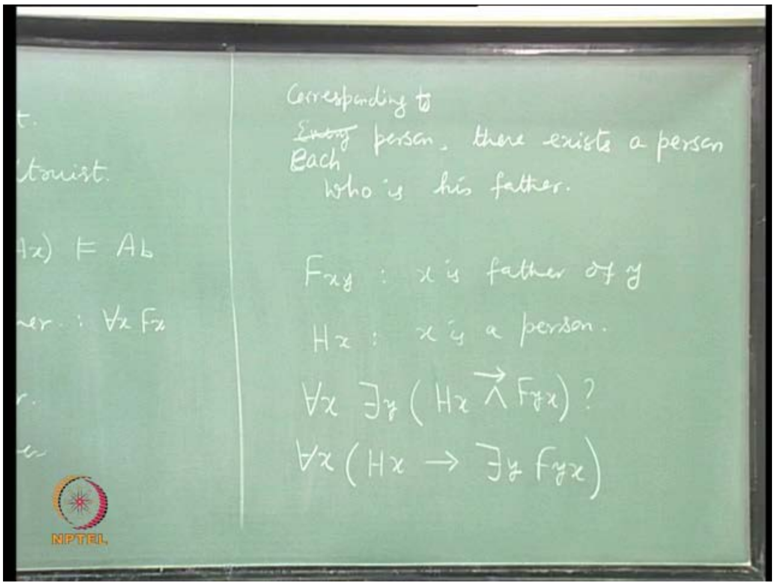
Now, what happens here, each x has a father, we will write as another predicate. Suppose x has a father, we will write as Fx.

Student: (Refer Time: 18:27) In Fx, is it 'has' or what, whether a predicate or F is a predicate and x is the….

F is the predicate, x has been used as an argument there. When you say 'has a father' with the blank, that is the predicate, right. When you fill up that blank it becomes a sentence. Say you fill up with Bapuji. Bapuji has a father; it becomes sentence. But if you write a variable there, it is not a sentence; it is an unfinished sentence, right? It has to precede with a quantifier; otherwise, it does not give us any meaning, is it clear?

Now, say that this sentence will be represented as 'for each x, Fx'. Suppose, we want to go a bit deeper. We say everyone means every man. Let us say, or every person now, then what do you say? 'has a father' means there is a person who is his father, right. So we can rewrite it in a different way.

(Refer Slide Time: 19:46)



That every person, corresponding to every person, there exists a person who is his father. Please do not get offended with his or her; I am using his for universal, we have to do something otherwise, we have to write always his and her.

Now, how do you translate this? This is also translation of the same sentence: everyone has a father. Now, how would you translate? See, 'has a father' is now dissected, right; it is no more one predicate. Somebody is a father, that is now a predicate, but it is not just is a father, it is his father. So, y is the father of x, that is really involved, right. It is not just 'is a father'. It is not a unary predicate. 'is a father', it is a binary predicate now. Somebody is somebody else's father. Now, we have Fxy; you may say x is father of y, add to vocabulary, and all this.

Now, if you take every person, somebody is a person, you want to go to that level, somebody may not be a person. So then, you have to introduce another predicate 'somebody is a person', 'is a man' or 'a human being'. For that you are keeping persons not for kangaroos. In that case we will say, let us say, H of x, x is a person. Now, how do you translate this? Corresponding to every person, so it is every, it is very ambiguous here. You read it as: corresponding to each person, then ambiguity is over; you do not say corresponding to all persons. Only God can come there, who is the father of everyone. So, you have to read it as: corresponding to each person, that is in the context. So, again you have to read it correctly. Better use the word 'each'. Then it says, whatever person you start with, correspond to each x, you will get another person, right, so there is y.

Student: Sir, first you said x is a person so Hx implies.

So you want to write first that x is a person. You will say: for each x if x is a person then there exists somebody who is father, so there exists some y such that

Student: Fyx.

Fyx. We wrote x, y, x is father of y. First one is the father of the second one. Here y should be your 'father of x'. We will write Fyx, is it alright? Can you write it this way also. Yes, there is a difference, the difference is, it says

Student: You said all x is a person.

All, x will become human beings. If you write like this, that will be the meaning of the sentence. For each x there exists one y who is his father, but what is that x? That x has to be a human being, right? Is that clear? It assumes that each x has to be human being here. Here, we do not assume. We say if it is, if it is not, then we do not know. Nothing is coded here. If x is not a human being, nothing is coded here. But x is not a human being does not arise here. That is taken. Every x, whatever object you take, that is a human being. That is the difference. Still, it can be a translation of that provided that way we do not know what it means. But common thing is this; this is what we understand immediately. And what about the sentence: for each x there is y Hx implies Fyx? But what about this?

Student: Same as below.

Is it same as below?

Student: Yeah.

Student: No, which means if at all for every y, you say that a y is a father of x.

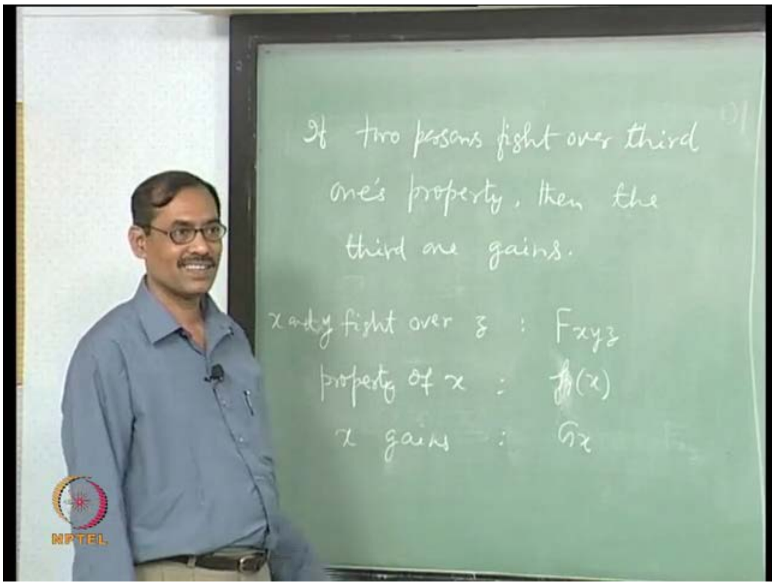
Student: Not for every y, only if Hx is.

Student: If x is a person for every y, you say that y is a father of x.

Student: He is saying here that if Hx is not true then definitely there exists a y such that there is a father; so that cannot hold.

Fy? We say not Hx or Fyx? Let us take it that way. For each x there exists y, x is not a person or it has a father. Think about it a bit. Are they same or not? If you write implies here and the last one, whether these two are same or not? We will come to it again. When we come to formal semantics, whether they are same or not. You remember and think about this. We will not decide about this now. We have not put two symbols now. There are two cases. One case is: with and, that we find it is not okey; it is not of the same meaning as this. Then you are putting implies and asking whether it has the same meaning as this, as whether it can be translated as that or not; we stop it there.

(Refer Slide Time: 27:43)

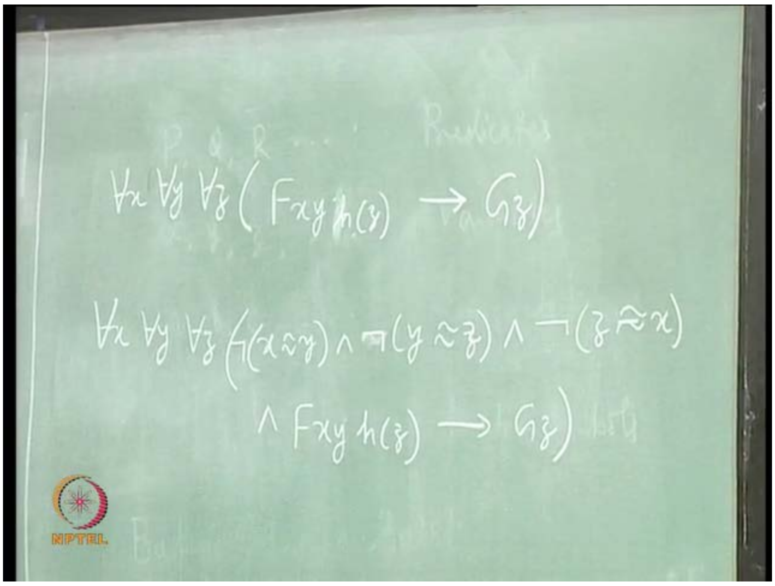


If two persons fight over third one's property, then the third one gains. Now, what is the vocabulary here?

Third one gains. We do not know whether it gains the same property or it gains somewhere else. It might be giving you a good omen, you will gain in some lottery. You can jut, if you do not want to go deeper, you can say 'fight over another's property' as one, but if you want to say 'it is third one's property' exactly not just another’s, then “somebody's property” will be one predicate. 'fight over something' and who are fighting that will be another predicate, right. Then 'gains something', that is another predicate. Just gains, x gains, that will be another predicate. Is that okey? Let us start with that. Say, 'fight over x and y, x and y, fight over, fight over z. Something, they are fighting over something, that may be property of somebody, fighting over z. So, this is now a ternary predicate. Let us write Fxyz.

Next, what is involved? Third one's property, someone's property. We will write x is someone's property. You want to write it as a function, property of x; that is better. So, property of x will be written as p of x you can write f of x, h of x. We will write g of x. But g will be used here, so write h of x. Next, x gains is a predicate. it is a unary predicate, we will right Gx. This is not for some; first one, some second one, and some third one. This is for every one, right. That, we have to check.

(Refer Slide Time: 31:54)



From this class, once you write, if 2 persons; is it taking only 2 persons from this class; or somewhere else? If nothing is mentioned like man is mortal. You do not say, each man is mortal, you just say, man is mortal. Then how to translate it? Whether some man is mortal or every man is mortal? Common sense says that every man is mortal, that is what it means. By the sentence man is mortal, right, it is a generic one. So, here, all those persons we can quantify, for all, for each, over all those persons. We may say for each x, for each y, for each z.

Student: It is not equal

Equality predicate will come later, slowly. Let us see first, this says Fx and z, if x and y fight over z; but they are not fighting over another person. You want for each person, here, right, they are fighting over another's property, so property of z let us say. You may write here H of z. Then what happens? The third one gains, Gz. that is all. Now, we come to the exact meaning. It says, that is first one, second one, third one. There are two persons and then the third person. If we write just like this, it will say one person can fight with himself, with his own property, that is allowed and he also gains. So, you do not want that vacuousness to be present, right? So we want all of them to be distinct, so we can use the equality predicate, that means x, y, and z should be distinct persons here.

Then you may say for each x, for each y, for each z, if x is not equal to y and y is not equal to z and z is not equal to x; we have to be very specific, right; and Fxyz then Gz. This is better than the earlier one. What about that earlier example? You have not told about that? Can one person be his own father? It allows.

Student: The statement allows sir.

No, common sense does not allow it. You have to take the common sense into count.

Student: Sir, father can just mean something, I can define father.

No, it can be anything, yes, see, father may not be father in common sense. It can be brother, it can be one person who is of the same sort, we do not know the meaning of father, in that sense it is alike. It can be any predicate, but it is not the one what we say in natural language. Once you say you have some meaning to attach to it.

Student: You are not saying for all x and for all y, that there exists y that day it was satisfied x is father of y.

Yes, that is a question. See the question here is: let us say last one, which we agreed to be alright. For each x, if x is a person then there is a person y who is his father. That also allows that x to become his own father; it can be, that same y can be that x also, it does not mean it has to be. Yes, somewhere that we have to say, if it is a person then there is another person who is his father; so then, there exists y such that y is not equal to x and Fyx.

Student: Fxy is a definition. We can say x is a father of y and x is not right.

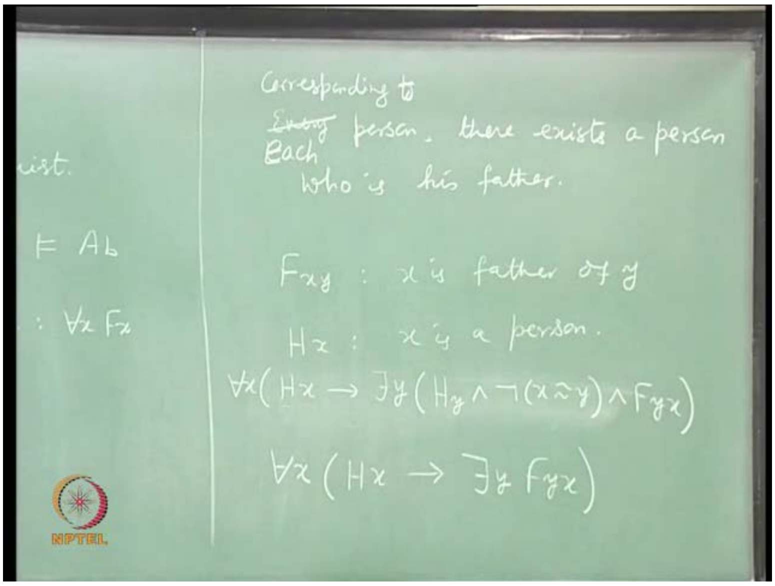
And x is not equal to y.

Student: Yeah, we can do that; that is easier.

But that will not allow you to write at other places, but that is alright, in this case also. You can include that itself here: x is father of y and x is not the same as y; included, but then it is better to include here. There, can be somewhere, you need x not equal to y.

Student: Here, we should also include that H of y in the implication, because it says there exists a person who is his father that is not implicated in Hy.

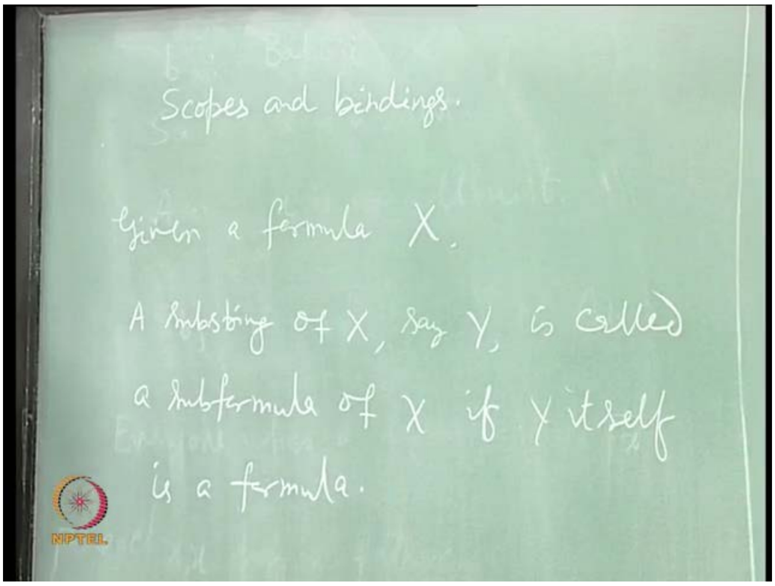
(Refer Slide Time: 36:41)



So, what will be the better translation here? You would say: for each x, Hx implies there exists y who is a person and not equal to x such that Hy and Fyx. It gives better sense than the earlier. Sometimes, such implicit assumptions will tell you that your argument is not correct. There may be there, some implicit assumption; that is why you are not getting the answer. So it has to go into the translation process itself. Common sense is the most difficult one and you have to tackle it at the time of translation itself, not later, is that clear? These types of things can come up in translation. You have to be cautious about it, that is what it says. Sometimes you need equality, sometimes you may not need, and so on.

Let us go back to our original thing. You have done some translations. The question is, how to say that whether something follows from it or it does not follow? And there is another thing that you might have noticed in all these translations it is not a sentence of the type p x, a sentence always will be coming for each x Px or there exists x Px, from natural language when it comes as a sentence it never comes like that. So, you want to really distinguish between these types of things where all the variables that are used, they are quantified and where they are not quantified, and something before we want there; so we will define certain concept which will be useful later.

(Refer Slide Time: 38:45)



That is called the scopes and bindings. These are really scopes and bindings of the quantifiers which we are using. Once scope, it is a very simple thing like, you consider a formula and take an occurrence of a quantifier, then start from that occurrence of a quantifier, find out till how much you can go to make a formula, that is called a subformula. In fact you can generalize it a bit. You say that any substring of the same formula which itself is a formula is called a subformula of the original, that is easier to express.

Given a formula say, X, is a substring of X, say Y is called a sub formula of X if Y itself is a formula. It has to be a substring, a part of that, and also it has to be a formula, then you call it is a subformula. That means you take any substring that is not really a subformula. Let us take any example, we have done. For example, I start from this place, this is a substring, but it is not a subformula. I start from this place. I write up to this, this is a substring, but this is a subformula, right, by itself it is subformula, though brackets and other things are needed. Now we are using abbreviated formulas; so we can make it simpler by using brackets alone.

Suppose, you take from this one, right from the abbreviated formulas. Now you take this as a string. This string by itself is a subformula, you may say. But in the abbreviated form it is giving the trouble. Suppose, it is in the expanded form. Then what happens? Where are the brackets? You will be having brackets, so many places. Let us take one side, say, I have a bracket here, then I have a bracket here, then I have a bracket here. Now, with the brackets you see it is not a subformula, is it clear? You have to be a bit careful whether brackets are there or not. But we are thinking always, not of the abbreviated formula, of the formula itself; whatever is the correct one, because subformula really needs your unique parsing. Unless you have the unique parsing, you cannot define subformula correctly. So, brackets are to be inserted to verify whether it is a subformula or not.

Let us see. For example, I take this as a, let us write it in the abbreviated form. It looks like: for each x for each y for each z not x equal to y not y equal to z not z equal to x and Fxy h of z implies g of z. Now, let us take one occurrence of the quantifier, say z. Now from starting from this place, you find a subformula. That is called the scope of this occurrence of the quantifier, is it now clear? Again you have to think of the abbreviated formulas and the formulas. If you have abbreviated formulas, there can be confusion of getting the scope. So, always you have to think of the formulas in the expanded form, in the correct form, not in the abbreviated formulas. But here it is easy, it is not confusing. It says that the scope of the formula is this underlined thing, scope of this occurrence of the quantifier.

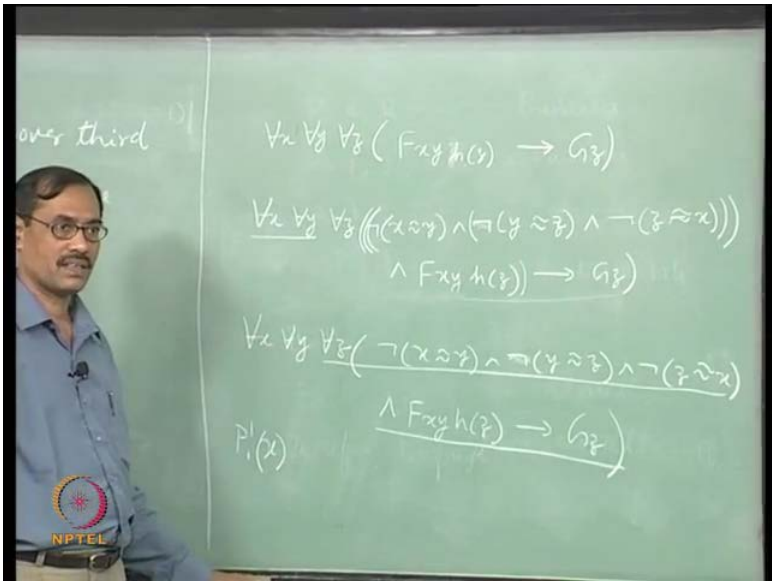
Student: Is it just a formula?

It is just a formula.

Student: But x and y are not quantified.

Does not matter. We have never told anywhere in a formula, definition of the formula, that everything has to be quantified. They are not sentences. When you translate from English sentences you will not get them, that is what it is, right? But they are allowed as formulas like in the atomic formulas. Suppose I have x as a variable. Every variable is a term, right. Now, suppose I take P 1 1. So P 1 is the predicate with one argument vacant. I have to do this; now this is a formula, by definition. Yes?

(Refer Slide Time: 44:09)



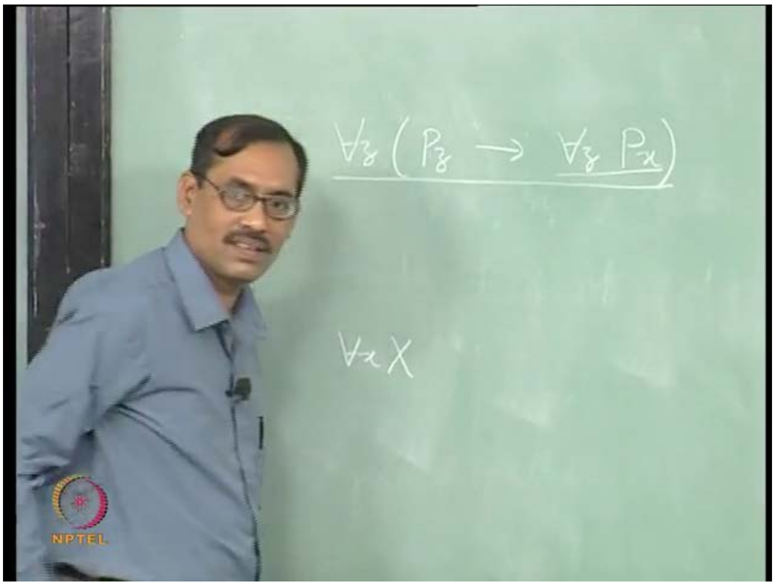
Student: Sir, where the formula is constructed you have never said, do not remove a variable. So even by construction you cannot really get that formula.

Well, how the terms are defined? If you go back to the terms, it says you have the variables, you have the constants, you have the function symbols. Using those things terms are generated. By definition, each constant is also a term, each variable is also a term, and then functions where arguments can be other terms; they can be variables, constants or other terms. And in the formulas, when you define, say, P of t1, t2, t n, you can have variables there, is that okey? So, this is allowed as a formula, though you have got it while translating from the English sentences. That is what we are saying here. You just construct a subformula starting from that occurrence, that gives you the scope of that occurrence of the quantifier.

Student: Scope is only for the quantifier.

Only for the occurrence of the quantifier. See, quantifier is only one here, for all, right? But there are three occurrences of the quantifier: one is with each x, for each y, with each z. It can happen, for this each z, also occurs somewhere.

(Refer Slide Time: 46:00)



Let us see an example. Suppose we have for each z P z implies for each z P x. This is also allowed as a formula even though z is not here. 'for each z' is there, still it is allowed, right? Because, all that we say is, for each x X if x is a formula, and small x is a variable, then for each x X is a formula, right? So, in this capital X we never said that small x has to occur. So everything is allowed there, because at this stage of the grammar we cannot specify all those things, but later we have to take care; at some point we will see what happens. Why we have allowed all those things, right. But till now it is allowed. Therefore, this for each z there are two occurrences; for each z, not only for each, for each z, there are two occurrences.

If I say what is the scope of `for each z', it will have no meaning. I have to say which occurrence so I say the first occurrence of each z, what is its scope? It is the whole formula. If I say what is the scope of the second occurrence of for each z? It will be only this much is, that clear? It is the occurrence of a quantifier and its scope, that is what we are defining, any doubts?

Student: Sir, scope includes the quantifier or not?

Yes, it includes the occurrence of the quantifier. It starts from this place itself, from that occurrence. Scope of for each z will be whole formula, right. One can define without it also, does not matter. But we will include it so that it will be easy for us to see; where, for what the formula it is giving rise to.