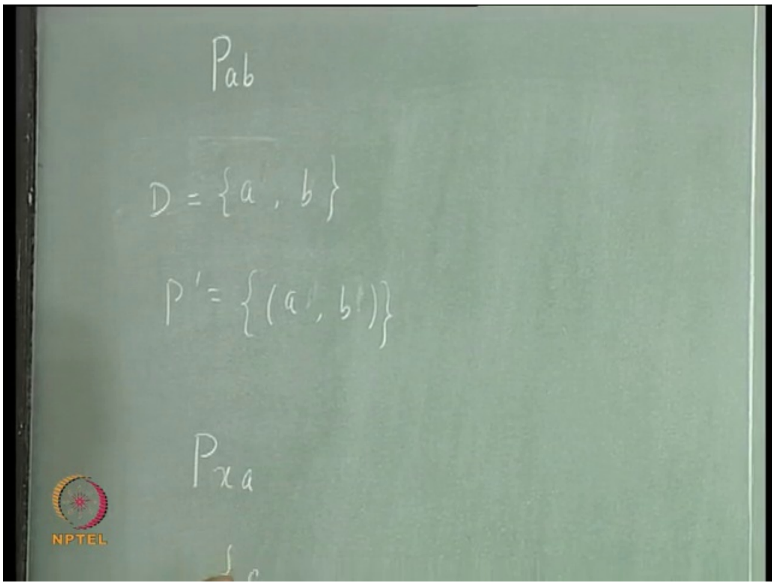
Mathematical Logic
Prof. Arindama Singh
Department of Mathematics
Indian Institute of Technology, Madras

Lecture - 34
Syntactic Interpretation

So, what we had done till now is, you start with a formula, then rectify it; after rectification, pull the quantifiers to the beginning, then use skolemization. Then after skolemization all the essential quantifiers are removed. Then all that you have, only possibly some universal quantifiers in the beginning; that is called the prefix of the formula, the block of quantifiers. Then you have the matrix of the formula which is having no quantifiers. Then convert the matrix to one CNF, right? You reach at SCNF, skolem conjunctive normal form, fine. Then we wanted to discuss something more on the SCNF's. Let us take one example; start with that.

(Refer Slide Time: 01:13)



Say, I have a simple formula P a b. This is in SCNF; it looks very trivial. But let us see; this will tell us what to do. This is in SCNF, fine; there is no quantifier at all. Now, a question is the SCNF is a form which preserves only satisfiability of the formula. If X was originally satisfiable then its SCNF is also satisfiable; and conversely; fine. Suppose this is our formula and this is also; it is SCNF.

Now, to say that it is satisfiable what are you going to do? Obviously it demands a model. If there is a model; it is a sentence also, all these SCNF are sentences; so, you need not consider the valuations at all, right? We can only think of interpretations, where there is a domain, and there is one map which assigns the predicates and function symbols to predicate, some functions on the domain, fine. Now, this formula; to say that it is satisfiable, what we will do?

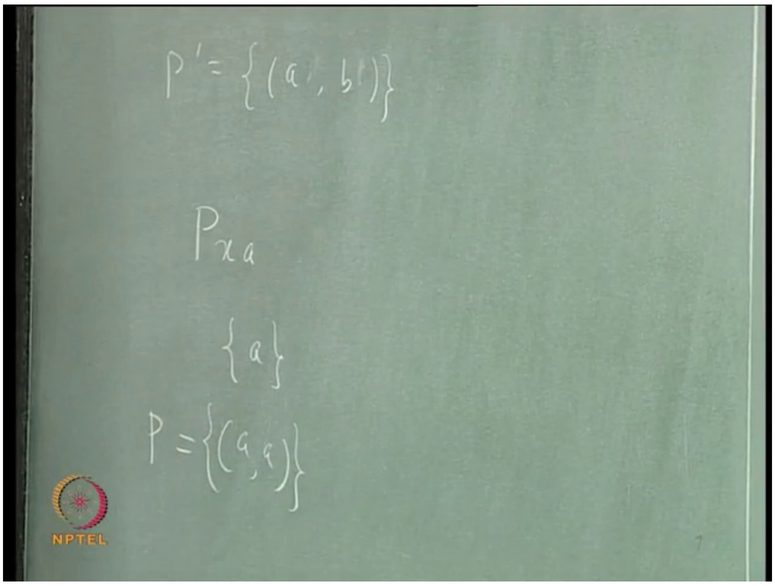
We can think of any domain where these constants a and b will be interpreted as elements of that domain and P will be interpreted as a binary relation. We should take natural numbers. So, you say a means it is two, b is three, and P means less than. Now, you have a got a model, right. But so many other models can be there. In fact there are so many possibilities; we do not know which one to choose. There are infinitely many possibilities. Any set you can choose and then try to define any relation and assign these constants to any elements in that domain, right. Choice is too much; we do not know how to make it mechanical. But there is a natural way to go for this mechanical model construction.

What we do, we have this a and b; let me take only two elements set, say, a prime, b prime. Where, implicitly I am taking that a is assigned to a prime, b is assigned to b prime, in my domain D. Then how to read this P? I have only occurrence of P a b. So, I will write my P prime as the set containing only ordered pair a prime, b prime. That is all. It is a model of this. Now, come to the thing that why to take this a prime, b prime? Why not just a, b? They are also symbols; because in this set, it is not a concrete set like one two and so on, having no definite properties. So, I can just as well take a, b forgetting this prime. And then my P will be a, b ordered pair. Now, why to assign even a, b ordered pair? I just declare P a b is true. So, I just get a model.

That means given P a b, I just think of a as an element of the domain, b as an element of the domain, where P a b is true. But this, always, we may not be able to do, because we cannot just say that whatever formula is there, it is true. Its structure is so, that it need not be true; like, for example, P a b and not P a b. Whatever a, b, I choose, I can never make it true, right? But I can always declare the atomic formulas to be true or false, not necessarily with connectives. That works. It can be done, right?

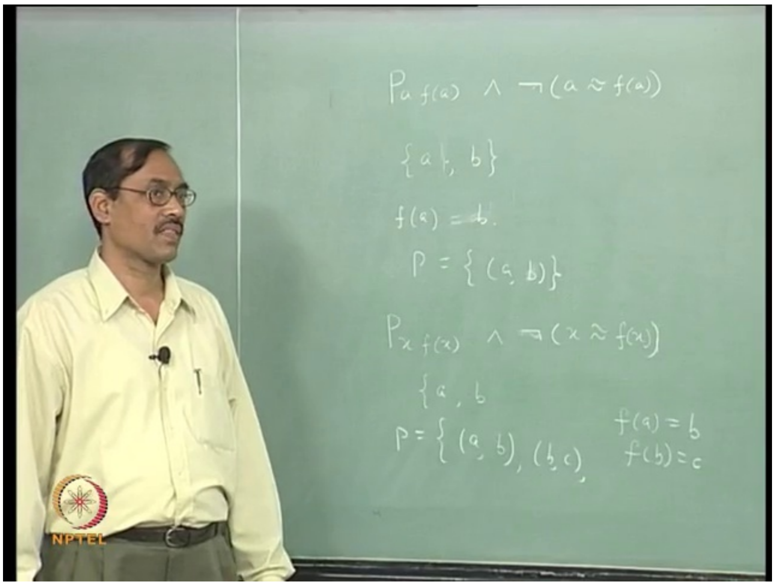
Let us take one more example before proceeding. Say, I have P x a. Now, how to construct a model? Well, this means for each x P x a. Once it is in SCNF, we have taken away all the quantifiers; they are all universally quantified. Now, all the free variables. So, this will say for each x P x a. Now, if you want to have a model or a domain, where it is an infinite set, then you have to give P, every element, then a, whatever is assigned, all those should be true. But who asks you take infinite? You can take also a finite domain. I can start with a itself.

(Refer Slide Time: 05:36)



So, only one element in my domain, a; and then I declare that a, a is having only element in P; that will make it a model. This will be really P prime, this is a prime, and so on. You are now forgetting that primes, fine. So, easily you can make.

(Refer Slide Time: 05:49)



Now, suppose it is P a f of a. Now, how far we can go, without constructing one domain? Well, you can think of one element a, where f of a is also equal to a, right. Then you can have P as the singleton, where a a, that is fine; that is allowed, right. But, suppose I have P a f a and a is not equal to f of a. Then I cannot take this f of a equal to a, because equality predicate has to be interpreted as equality in strings, or equality in the domain. Whatever is the domain.

Now, we do not have any definite domain; we are just taking the symbols a or only equality, there is equality as strings. They are just formal symbols, right. So, I cannot take f a equal to a. As strings, they are different. I cannot say a is equal f a. Then what to do? Take another b. Now, you define f of a equal to b; that is fine, no problem. So, just define f of a equal to b; that gives a model. So, P has to be changed; and this will b, and then a is not equal to b, in this domain, fine. That gives a model; is that clear?

But, always, that also may not succeed. Because, suppose I have P x f of x. Now, suppose we start with a in your domain. It needs another which is f of this, say, along with this, I have not of x equal to f of x. So, we will have in P, at least a and b. So, I give b; and then we have to define f of a equal to b, right. But this should be true for every x in the domain, right? What should be there again for b? I have to define something. So I say, define f of b equal to c. I give b, c. Now, it becomes infinite, right? f of b equal to f, we can put.

Student: But this may not give a model.

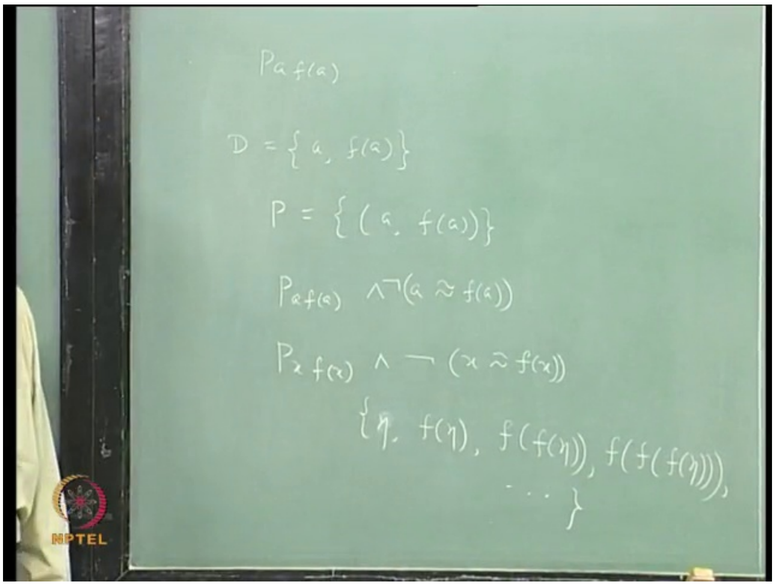
Yes, why not.

Student: If there is negation and with equality?

Negation of x equal to f x. So, a is not equal to f of a, then b is not equal to f of b. That is clear, right? But f of f of a can be equal to a, that does not put any restriction. So, still it can be done, right. But there is a problem. We are now varying this b; they are no more looking like strings as per our convention. Here, for a, we are just writing as a. Now, for f of a, we should write f of a, not b, right? So, if you proceed that way, what do you get? You will be getting one infinite set, because equality will be equality of strings only, in that sense no more equality you will be defining, right? If that is our proposal, then it will end up in an infinite set. It will not stop, yes?

See, here when we take, that we will write with primes. But consider them as elements. That means a is considered as a itself, b is considered as b itself. Now, when you come to a, f a we will taking a as a itself, but we are not going to define f of a equal to b anymore; we are taking f of a itself, right? Instead of f of a written as b, we will be taking f of a itself. Once you do that, you cannot write f of f of a will be equal to a again. They will be different as strings, right? So, that will end up in one infinite set, right. But what is the advantage in doing this? It may be possible; always I will be able to do this; but I can suppose I have the formula and x is not equal to f of f of x, then this will again take another element. So, in general you may not be able to get always this finite model. There are, in fact, formulas for which there exist no finite models; only on infinite sets they will be true, right? So, in those cases this kind of model building will fail, fine? We have to go back to our earlier scheme.

(Refer Slide Time: 11:22)



But instead of taking is as b, let us write it as f of f. In that case, for example, this even P a f, we would build a domain as a, f of a, and you can stop there, right? Then declare that P will be equal to a and f a, ordered pair only, nothing else; they are related by P; that is all right. In case of P x or even P a f of a and a is not equal to f of a? In this case, what should we do? I cannot take a, f a, but a, f a will do? It is not equal to f of a, still. Again, this one domain works fine. If it is P x f of x and x is not equal to f of x, then there might be problem, right. I may start with a. So, a does not occur in the formula right. Let me give some name instead of a, call it eta. I just start with some symbol eta; then I need f of eta, right. But to make this true, I need f of eta as x and this becomes f of f of eta, fine. I need another element f of f of eta. Again by taking f of f of eta here, I need f of f of f of eta, right? So, you put that again. Now, it continues, it becomes an infinite set. Fine. We can take the infinite set.

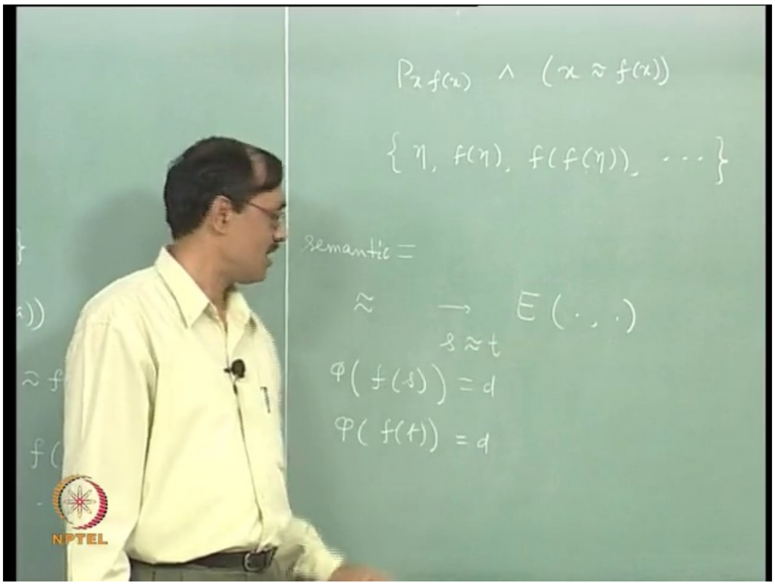
Student: We say that f of a is not equal to a by definition?

We are not even defining it; as strings they are not equal.

Student: Yeah, strings are not equal, but.

But it raises another question. We say, suppose, instead of not equal to, we have equal to, what will you do? this is the problem.

(Refer Slide Time: 13:59)



Not equal to is already satisfied, because as strings they are not equal, fine. But now when you have equality relation, say, P x f of x and x is equal to f of x. I can have the same domain again, eta, f of eta, f of f of eta, and so on. Now, how to tackle this x is equal to f of x. So, for P, I can define as it is. P of all x f x. So, that f belongs to this, right? That will be my set. Now, for equality, I cannot write x equal to f of x, because as strings they are not really equal.

So, equality relation has to be tackled separately. It is not just writing P something. All this P's can be interpreted later; we can give them zero or one. For each of the elements in this domain, formal domain, I can declare P of eta, f of eta is 1, P of f of eta f of f of eta is 1. Declare them; all the others you take them zero fine. But then what about the equality? We have to do something here. Equality cannot be just the equality of strings; that is what it says.

Student: So, here we are doing it, sort of syntactically; and there we are, we have to find a semantic.

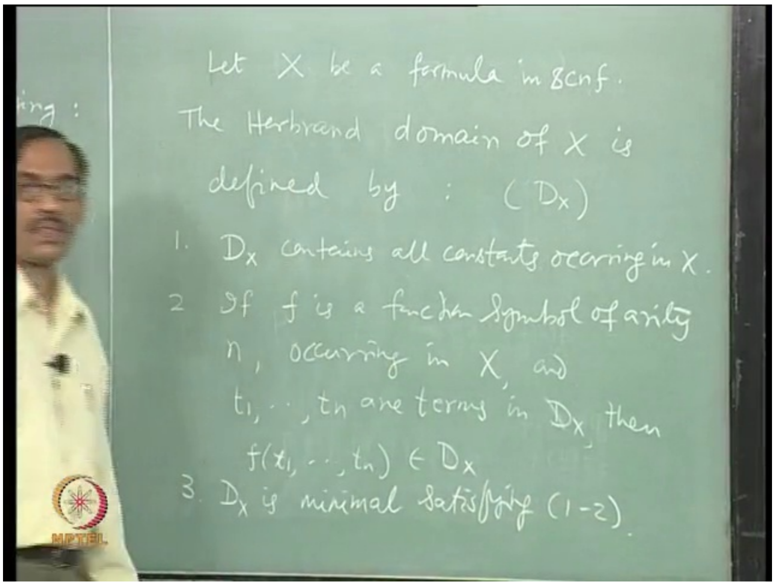
You have to do something there. It is not exactly the string you call it. You have to give another definition of equality there. It is not just the strings equality there; some other equality will do. But what equality will be appropriate, in the syntactic domain? It is a syntactic domain; you are just formally making the domain without giving any structure, without bringing any set, no numbers, no patterns are there. Only except generated from the terms, generated from the function symbols.

We have to find out what properties of equality we are interested in. That will give us a definition of semantic equivalent of equality in this formal domain. Since it is equality relation which was preserved for equality in the domain semantically, we need some properties of equality relation on the domain; semantic equality, which we are writing as equal to, this is really semantic. So, some properties of this we need, which we are interested in. First thing is, whatever is that relation, let us call that E. Suppose this is related to, interpreted as, relation E; it is a binary predicate. E also will be a binary predicate. It is a binary predicate.

Now, this E will have some properties of equality; not all. If all, then it will be that string equality, which is not possible. Which properties we are interested in? First thing is, you may take that as: it is an equivalence relation. Because equality is an equivalence relation. It is reflexive, symmetric and transitive; that much should be there. Other thing we need is, substitutivity. Suppose, in a term you substitute one equal to another. Say, s is there occurring, t is also occurring, and s is equal to t. That is what we have to find. You will find in the formula that s equal to occurs. Then you should be able to substitute s for t and its valuation later should be same; is it clear? What I am telling, suppose f is a function, where s is occurring and you have a formula s equal to t. Then f of s is assigned by phi to some element say d. Then also with the same phi you should get f of t equal to d; that property we need. Now, this equality we are not having anymore; there is no domain here. All that we can say that these two elements should be related by this E. If s and t are related by this E, then these two also should be related by E; that is the substitutive property in terms.

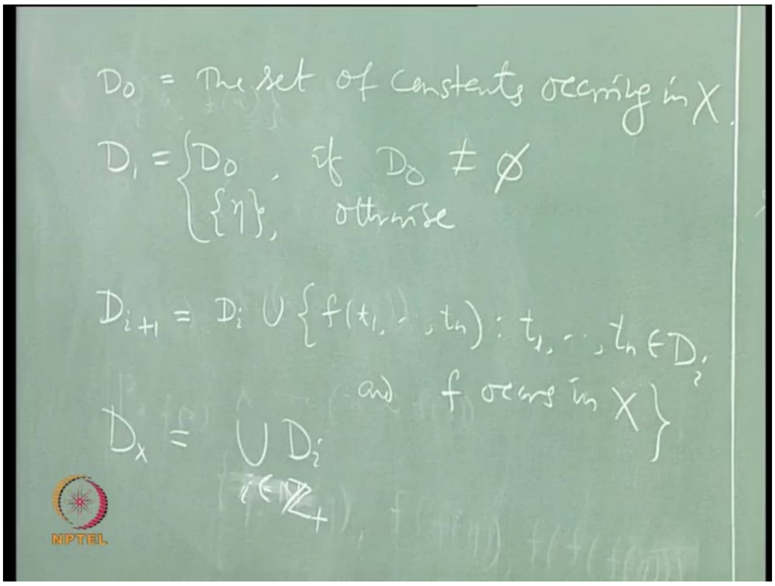
Let us write it now. For the equality relation, equality relation we are not writing, this E. So, E is a binary relation satisfying some properties. These are the properties it will satisfy. First is, E is an equivalence relation. Now, the question is what is the domain of this E? We have not specified it formally; but we have explained enough. That will start from this syntactic domain; where, you start from all the constants in the formula. If no constant is there, then write some eta. Then, if there are functions, you generate all terms from this, using the functions, right. That is your domain.

(Refer Slide Time: 19:43)



So, let us write it formally, if you are not getting it. Let X be a formula in skolem form, or let us say SCNF. Now, we are going to define a domain D. Let us give it a name; it is called a Herbrand domain. Herbrand domain of X is defined by, we should give some name, say it is D x. So, D x contains all constants occurring in X; that is the first rule. We should have, we should start from the constants. They will be interpreted as themselves. Second, if f is a function symbol of arity n, occurring in X, in the formula X and say t 1 to t n are terms in D x, then f of t 1 to t n belongs to D x. It should be closed under taking function symbols, right, forming the terms. Then, third thing is, it can be anything satisfying these two. We do not want that. We want it to be minimal; nothing else is there. It contains only this kind of terms. So, D x is minimal, satisfying 1 and 2. This is our domain, we are considering. Now, one question is, if D x does not have any constant, that is possible. In that case, we start from eta. That, we have to specify there. If no constants occur, if none in X, then D x contains eta. Some symbol we are using; let us call it eta. That means you will be starting the domain from the constants.

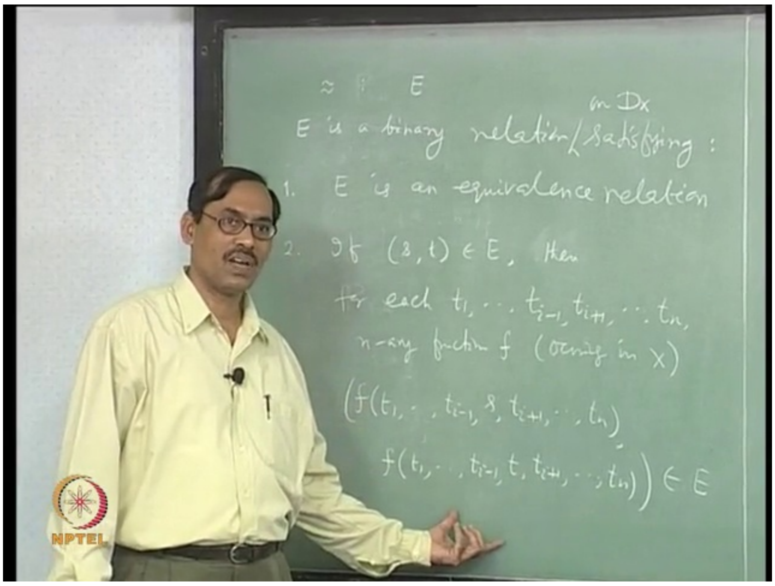
(Refer Slide Time: 22:58)



So, what we do is, take D 0 equal to the set of constants occurring in X. D 1 equal to D 0, if D 0 is nonempty; otherwise it is eta. When D 0 is empty, we just start with eta. Then D i plus 1 equal to, we say, D i union f of t 1 to t n, such that t 1 to t n belongs to D i and f occurs in X. Then your D X will be equal to union of all these D i's. We do not need to take D 0 anywhere; we can just write i belongs to Z plus. There is no harm in taking D 0 also, right? It is contained in D 1. That is how we will be proceeding to get your domain. In that domain only, we are considering this equivalence relation. So, our interpretation, our model, what we are going to construct, will be on this domain D. Sometimes you just write D instead of D sub X, if X is clear from the context.

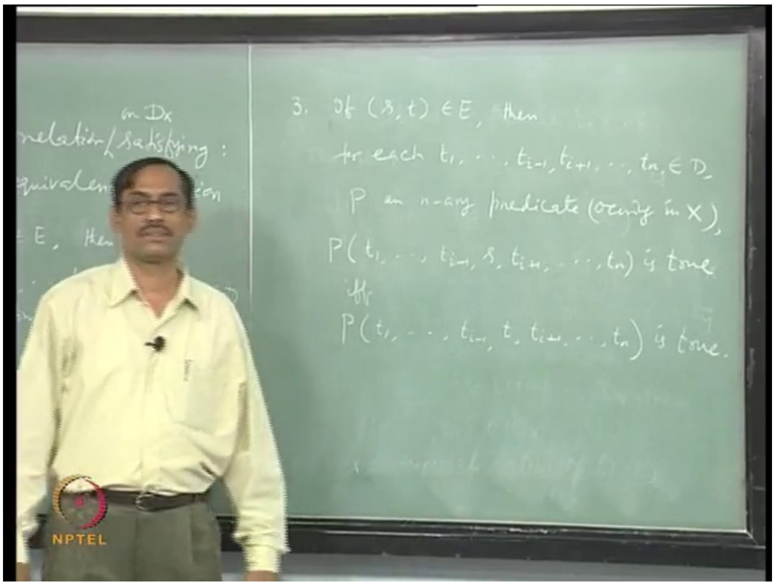
So now, this E is a relation, a binary relation on D, or let us write D X. Once a formula X is there, it will be D x; the Herbrand domain D. Now, this is also written as H sub D instead of D X. So, E is an equivalence relation. Second is, there should be substitutivity property for the terms. We will write that. If s comma t belongs to E, then that means s is equal to t. That is our sense of interpretation here. If s and t are related by this equivalence relation, s t belongs to E, then? Now, we need also terms. You may write for each term, there is no problem. Then for each t 1 to say t i minus one, t i plus 1, to t n and n-ary function f, you may say, occurring here in X; because they have the terms only in D; f of t 1 to t i minus 1 s t i plus 1 to t n should be equal to f of t 1 to t i minus 1 t t i plus 1. So, t I, that means this other pair must be in this. That gives the substitutivity property in terms.

(Refer Slide Time: 24:45)



Once s is equal to t, you can substitute and say that this term is equal to this term. In fact that equal to will be in the domain, in the interpretation sense. Whatever element this is assigned to will be assigned to this alone. Now you have the domain itself. We just write, this belongs to E. One more we need; which is for the predicates. We cannot write predicate, ordered pair of the predicates belongs to E. Either a predicate will become to true or it is false; by the interpretation.

(Refer Slide Time: 27:44)



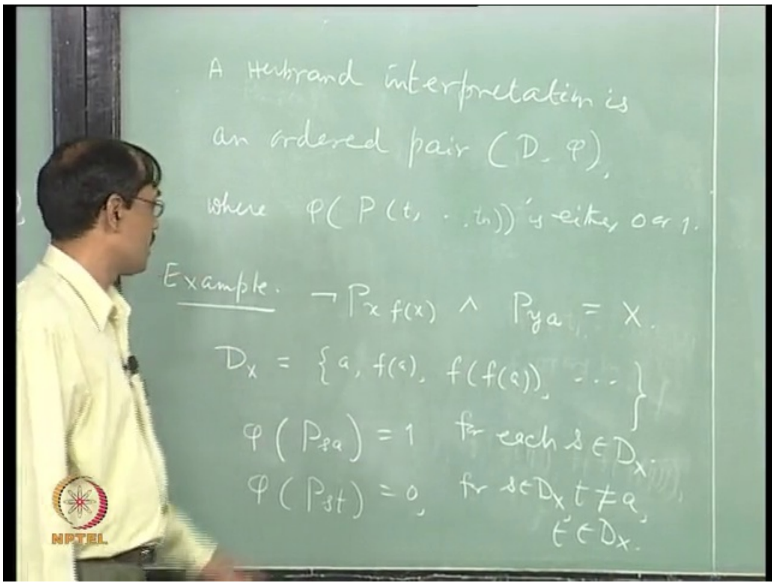
So, what we do, we write similar thing but now with truth or falsity. If s t belong to E, then for each t 1 to t i minus 1, t i plus 1 to t n, which are terms; these are in D and P an n-ary predicate occurring in X. What we have is, P of this t 1 to t i minus 1 s, t i plus 1 to t n is true if and only if P of t 1 to t i minus 1, t i plus 1 to t n is true. Even you can write if then, that is enough; it is an equivalence relation. For t for s or s for t, does not matter. So, this must hold. These are the only properties of E we have; we do not need equality. But at least one binary predicate which will we have this way, that will be our analogy for equal to.

Now what one Herbrand interpretation will do? We want interpret, right? We have a domain. Now, all the predicates are just written like relations. They are remaining with the same symbol; but we are now thinking them, assume, they are relations, not predicates; all functions give rise to terms. In fact they are closed terms only. After they are evaluated they will give rise to terms. They remain as functions themselves; the same symbols are used. Then what else we need for the interpretation? We should tell when some atomic formulas are true, when they are false; that much only we need, right. The phi, the second component phi, need not be specified for anything else; only it can say when an atomic proposition is true, when it is false. Once it says this much, all the other things are taken care by the connectives, right.

That means Herbrand interpretation will be an ordered pair, where D is this, the domain, and there is one component phi. That phi we need not specify, for how the predicates are interpreted, how the function are interpreted? Well, we need to specify how it assigns 0 or 1 to the atomic formulas. The atomic here will be of the form P of t 1 to t n. That t 1 to t n are closed terms; there is no variable in that. In this sense, these atomic terms are called ground terms and these atomic formulas, what we are getting, are called ground atomic formulas. Along with that their negations, called ground literals. So, everywhere one ground is added as an adjective there. Now, what we do is, just define the Herbrand interpretation.

A Herbrand interpretation is an ordered pair which is D phi. In fact, we are not defining Herbrand interpretation, in general. We are defining Herbrand interpretation for a formula X. So, this D is really D x. Once X is in the context, this D will be D x. Then this phi, where phi of any P of any t 1 to t n is either 0 or 1; it fixes in someway. So, phi fixes this to 0 or 1; something it fixes. Once it is fixed, your Herbrand interpretation is fixed.

(Refer Slide Time: 31:00)

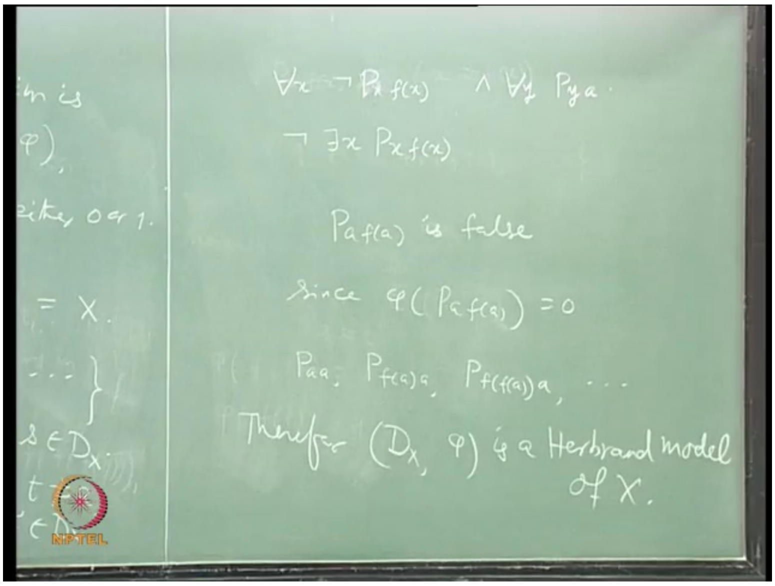


We will see an example, how this goes. Let us see some simple example. We should start with, say, not P x f of x and say P y a. Let us start with this; that is our X. Now, our Herbrand domain, which is D x, we will write as a only. But there is function symbol, f it starts from a. Now, your D 0 is having a, then D 1 we will have f of a along with a, D 2 will have a f of a f of f of a. So, it continues. Once a function symbol is there, it will be infinite. We take a, f of a, and so on. That is our Herbrand domain. What is the Herbrand map phi? We will call this as Herbrand map, which is fixing in the Herbrand interpretation. That Herbrand map phi; let us fix it this way. See, we cannot do it as it is. Because you have to go for the ground instances. When put this ground terms there, what formulas you are getting? Here it is easy to see. It will tell P a f of a P f of a f of f of a and so on. In general P f n minus 1 a f n a. It looks like that.

To all those things, we have to specify how this phi behaves. Also we have to say P, any y from this, a; how that also behaves, right? Suppose you want to make a Herbrand model which will satisfy all this formulas, then both this must be true at the same time. So, this must be true at least, this will be false, P x f x will be false P y a should be true. Now, if P of y a is true we have to write P s a equal to 1 where s varies from anything. Let us define that. Say, P s a equal to 1 for each s in D X. All other things we can take to be 0, does not matter; and one among them will be this, of this form. Let us take phi of P s t equal to 0 for s in D X and t not equal to a of course, t will have to be s.

Suppose we define like this. Now, is it a model? We have constructed for that purpose only; you have to see whether it is a model or not. Now, when you come to this formula, it is for each x for each y E. The whole thing, that is the really a formula, right? Because x and y are free variables; they are omitted from SCNF. They are really universally quantified, fine.

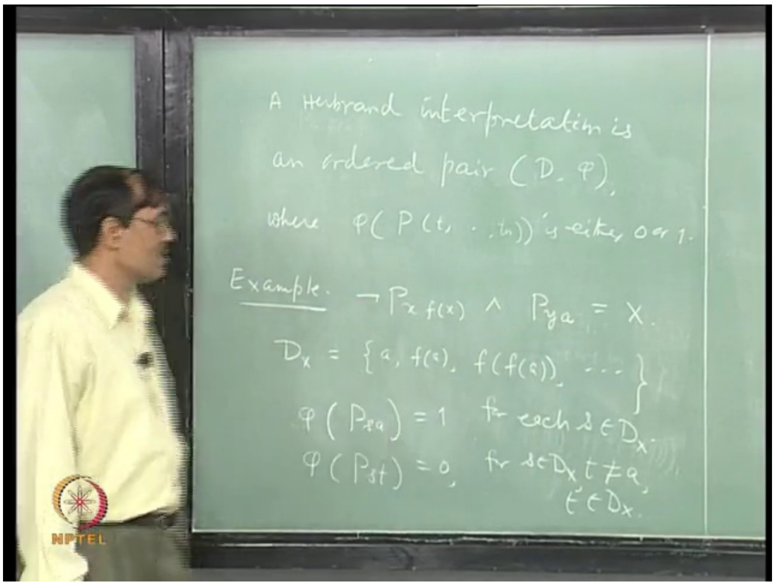
(Refer Slide Time: 35:50)



Now, SCNF we have taken, because it can distribute over the and, fine. This will be really equivalent to for each x not P x f x and for each y P y a; this is the formula at hand. Now, you claim that this D phi is a model of this; that is our claim, we want to verify. We will be going for first formula; let us say, for each x not P x f x, whether that is true or false. This is same thing as telling not there is x P x f x. This will be true when you find one element from your domain such that P x f x is false. For example, take a. We see that P a f of a is evaluated to 0, is false. Since phi of P a f a, in our definition, is equal to 0; that is our definition; whatever t is, if it is not equal to a, it will be 0. So it is 0. Once this is 0, it says, there exists one x which is equal to a, in our domain, such that P a f a is false. So, not of that will be true, right. So, first formula is true. Second one is, for each y P y a. That means we should have P a a, P f of a a, P f of f of a a; all these should be true. That is true, because P s a equal to 1 for each a, s in D X. So, you have got a Herbrand model.

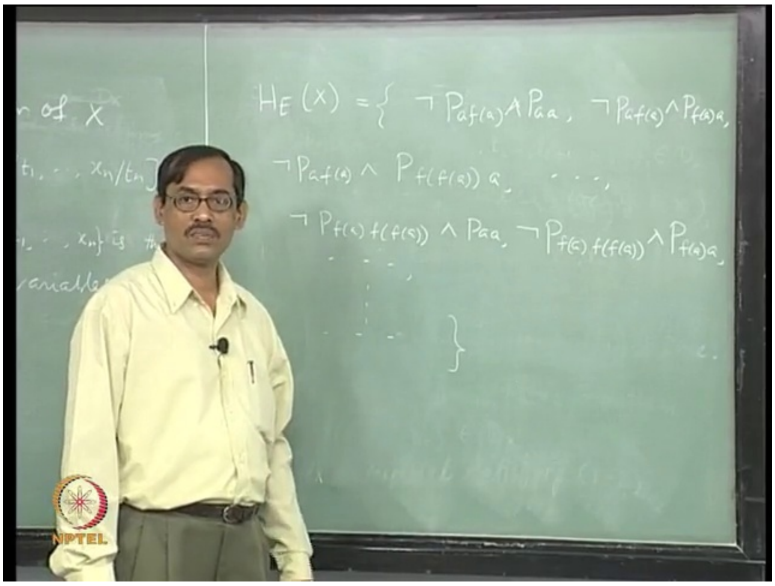
Therefore, D X and phi is a Herbrand model of X. This is the reason Herbrand is called the first computer scientist. He made it mechanical, almost; even for the first order logic, how mechanically we should proceed. To show that some first order formula is satisfiable, we do not have to consider any domain; now just proceed formally. This also gives rise to another thing, let us see in that example what is happening. See, I have these terms in D x: a, f of a, f of f of a, and so on; all these are ground terms. What we do, we just substitute in place of variables, all these terms, and get some formulas. There are infinite number of terms here; we will get infinite formulas from X itself. That means, substituting the free variables by the terms from D X, ground terms, we get the ground instances of the formula X, right. They are called the ground instances. The set of all ground instances, if you take, that is called the Herbrand expansion; is the Herbrand expansion of X. Let us compute that Herbrand expansion.

(Refer Slide Time: 39:52)



Here, we may define it. Herbrand expansion of X is, let us write, H E of X, which is equal to the set of all X, where x 1 substituted by t 1, x n substituted t n, such that t 1 to t n are in D X, x 1 to x n are all free variables, or rather we write this way. This is the set of all free variables of X. This is the Herbrand expansion. Each one of these elements in the Herbrand expansion is called a ground instance of X. Look at the Herbrand expansion of that formula X, in our example. Here, to get the Herbrand expansion what I do is, first I substitute x as a, right, and similarly, y as a, x and y can vary over the whole domain D. So, first I get not P a f of a and P a a, we just keep that a, x and vary this y. I must get everything. So, vary this y. That will give not P a f of a and P f of a a. Next, not P a f of a and P f of f of a, that continues, varying this y. Next, I will have, varying x again, not P f of a, f of f of a and P a a, not P f of a f of f of a, and P f of a a; that again continues.

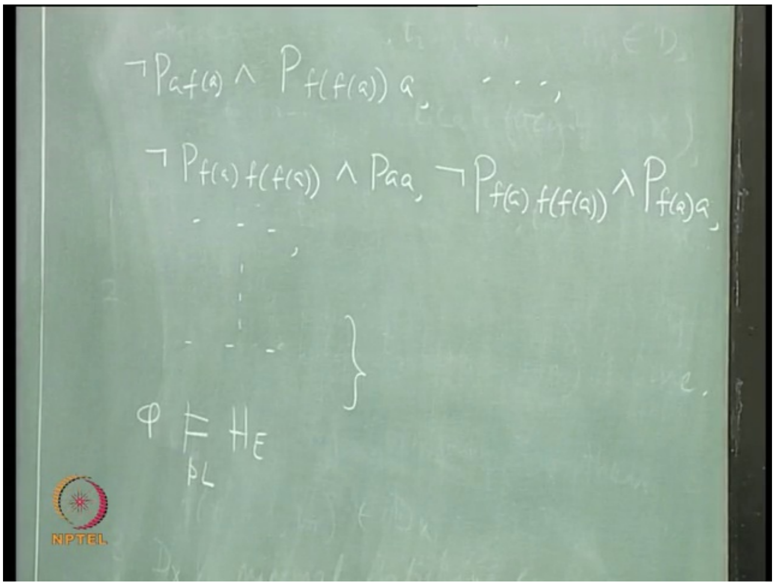
(Refer Slide Time: 41:28)



So, same way it continues. I find that infinite set; that is its Herbrand expansion. All possibilities for x and y, you substitute; get the ground instances; set of all those ground instances is the Herbrand expansion. Right? What this suggests is that whatever model you have chosen, phi of P s a equal to 1, phi of P s t equal to 0; this way. The same thing, thought as a propositional assignment, right, they are now just propositions; there are no quantifiers, nothing is there, atomic propositions, you can think of them. For these atomic things, if I think of this as a PL interpretation, propositional logic interpretation, then the same thing should be a model of the Herbrand expansion. Because, that is what basically you are doing: is it true or not, just verify, in this case. Here it says P s a equal to 1 for each s in D x. That is, P a a is 1. P a a is 1, and P f a a is 1, P f of f of a a is 1. So, all these and things, everywhere, they are 1; they are true now. Next thing it says: P s t equal to 0, for t not equal to a. So, not P s t equal to 1, for t not equal to a; not P s t equal to 1,whatever s t may be; t should not be a. So, if t is not a, they are all 1. Everywhere, t is not a. So, the whole set is satisfied now.

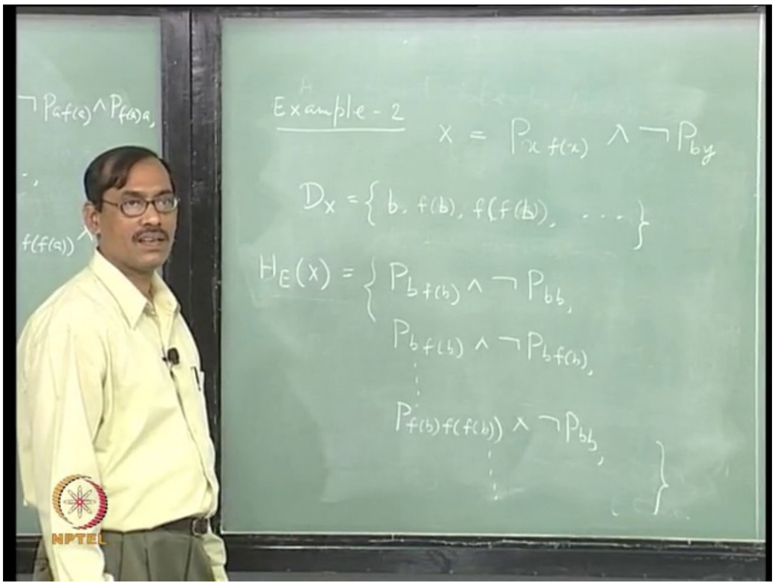
The same phi satisfies H E. What you see is that phi is a model of H E. This is a propositional model now. It propositionally satisfies H e. Right? So, what Herbrand did is that a first order formula is satisfiable if and only if some infinite set of, possibly an infinite set of, propositions is satisfiable; that is the contribution. You can reduce first order satisfiability to propositional satisfiability. Just we have to check propositional satisfiability here; but it might end up in an infinite set of formulas; that is it.

(Refer Slide Time: 44:37)



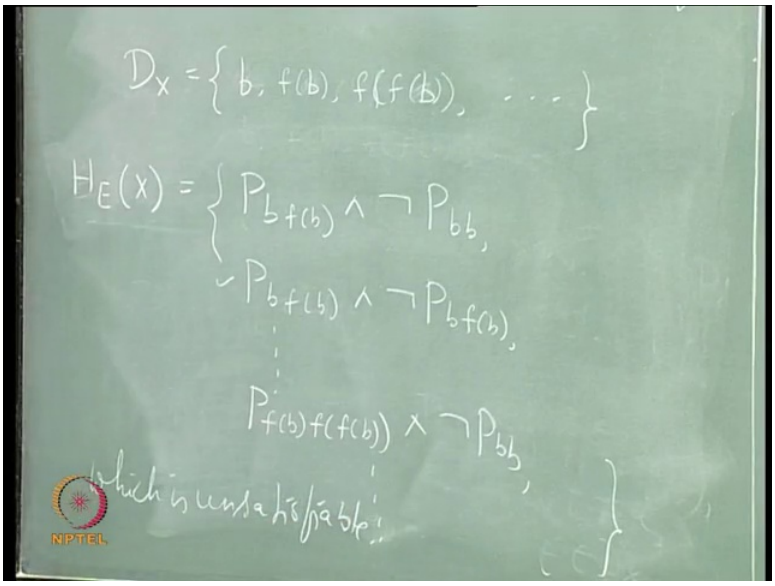
That is very clear, of course, once you take any particular domain. There, it is clear; in that domain you substitute all those things; satisfy for the relation; whether they satisfy the relations or not; that is again propositional. But what he did is, you do not have to search for any domain. You just confine yourself to the Herbrand domain; that is enough. Is it clear? But you have to give a proof of this, really, that H E propositionally satisfiability if and only if X is having a model, having a Herbrand model. Then you have to show: X is satisfiable if and only if X has a Herbrand model. Two things we have to see here. X is satisfiability if and only if it has a Herbrand model if and only if its Herbrand expansion is propositionally satisfiable.

(Refer Slide Time: 46:16)



Let us see another example. I have just changed a bit there. It should be easy to do. Here, our D X is equal to, say, a, same D x , but we have a b. So, you may have b, and so on. But there is no way here. So, instead of a, we have b; it is not another, and so on. Earlier there was a there is a, now only b is there. So, from b it will start, it will look like this. Then what is its Herbrand expansion? You just have to substitute x and y in place of this. Substitute x and y for these terms, ground terms. We get P b f of b and not P b b, P b f of b and not P b f of b. Then it continues: P f of b f of f of b or not P b b, it continues. This is the Herbrand expansion. Now, we see that it is not satisfiable; immediately, because second time, second ground instance, that ground instance is not satisfiable. So, this is not satisfiable. Because of this. It is unsatisfiable.

(Refer Slide Time: 49:00)



Now, you can show easily that this is also unsatisfiable. Can you see or not? You take any domain; b is interpreted as something, call it b prime, f is interpreted as something, some f prime. So, call it f prime of b prime, instead of f of x f of b. Then you just instantiate it, this. It is, for each x for y, everywhere; it should be true. But for that b and f b, it is not true. Therefore, it is unsatisfiable. The proof becomes also simple because of the Herbrand expansion. You can really prove how it is unsatisfiable, easily. So, the whole structure is same now, quickly, once you have the Herbrand expansion for this. This is what we are going to see next. That a formula is satisfiable if and only if it has a Herbrand model if and only if its Herbrand expansion is satisfiable.