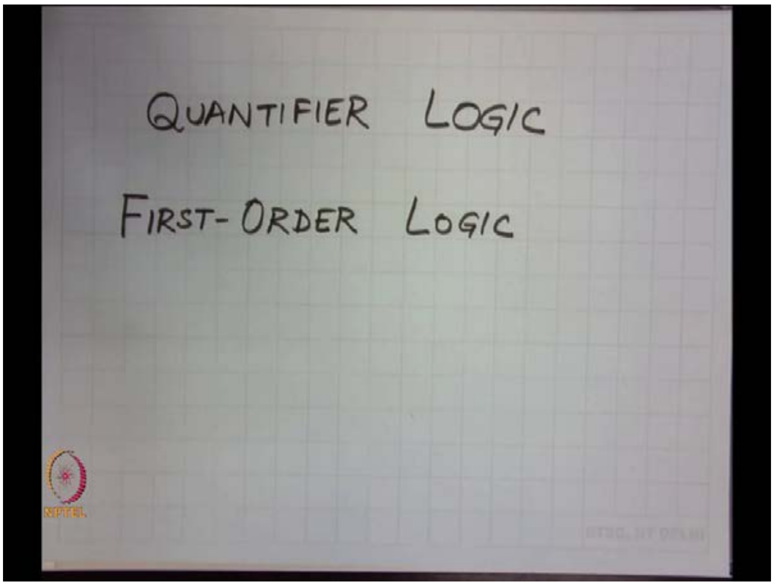
Logic for CS
Prof. Dr. S. Arun Kumar
Department of Computer Science
Indian Institute of Technology, Delhi

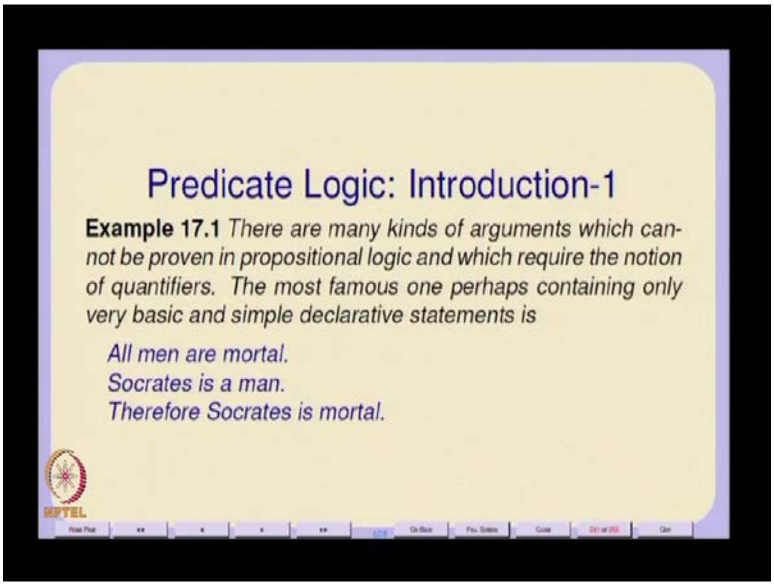
Lecture - 17
Consistency and Completeness

(Refer Slide Time: 00:39)



We want to introduce predicate logic predicate logic is also called some times its called quantifier logic and also its also sometimes is we going to study is we called first order logic and the question of what exactly the first order means is something that I will talk about later after we defined about the syntax may be. So, let us start with predicate logic or quantifier logic. So, so the important thing is to realize that there is there is something about or may be I will come to that as we proceed. So, the most famous valid argument since the time of aristotal.

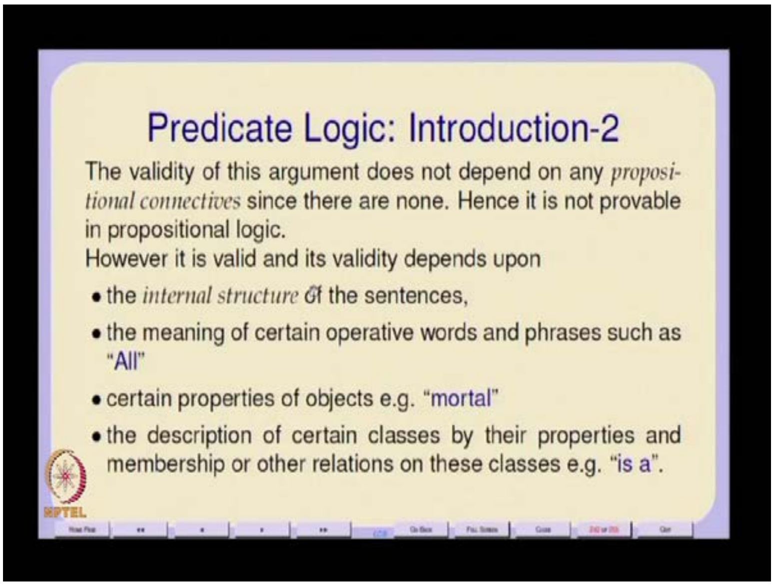
(Refer Slide Time: 01:32)



Which introduces the subject and might as well do this is this a famous thing that all men are mortal Socrates is a man and Socrates is a mortal. So, the interested thing about this argument is that all the sentences are very simple sentences there are no propositional connectives and and there is and and yet is not a valid argument and what we would like to know is exactly therefore, what is it is not provable in propositional logic since all of them are simple sentences you will give all of them separate propositional symbols and it cannot be broken up and cannot compound propositions that are simple propositions. So, essentially you will never be able to prove the validity of this argument from propositional logic right. So, so the there are other similar things, but what we should do is is instructive to just look at this this argument and see therefore, what is actually lacking in propositional logic ya.

one thing of course, is that the propositional connectives are more or less like exept for the negation ya propositional connectives are more or less like propositions in the language or conjunctions in the language. So, they are basically connect one or more sentences different simpler sentences may be and where as hare we are looking at something goes somewhat deeper in to it that is that. So, there are there are no propositional connectives visible anywhere you might transform the sentences. So, there is something else happening which is which we should look at and that is that.

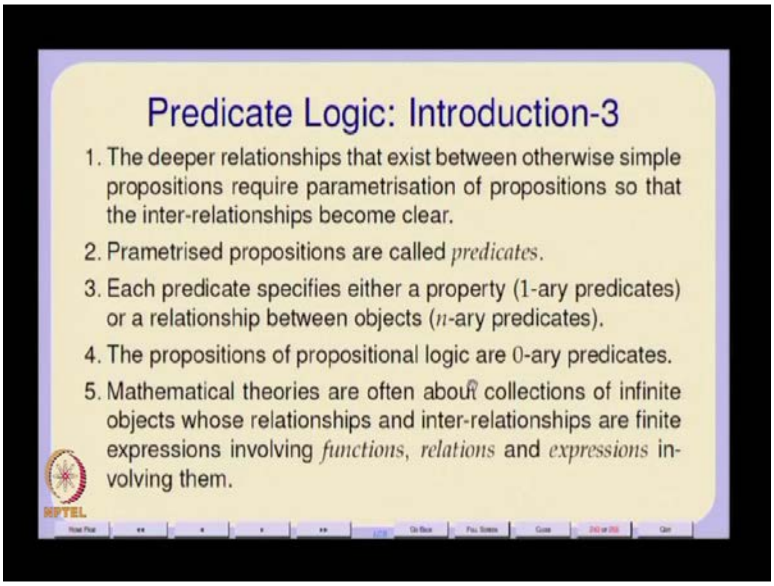
(Refer Slide Time: 03:41)



So firstly, what happens is that in this kind of an argument there is an even though all the sentences are simpler simple there is certain internal structure which is somehow being used remember that at a fundamental level we still want logic to be syntactic and based von forms, but the point is that propositional forms have been exhausted and then if you have to and actually have an expressive logic which can account for a large amount of the kinds of reasoning we do then we require to have a greater id better id of the internal structure of this even this simple sentences in what happens is that there are certain keywords like all and some and they are certain properties like in linguistic since they are like adjectives. So, they are not like propositions and conjunctions. So, there are properties of certain objects like mortal for example, and then there is a description of essentially classes of objects that is which is not there in which is not there in the propositional case which means that. So, that and then which means about classes when you are talking about classes and then you are essentially talking about membership in classes and you are talking about sometimes relationships between different classes. So, if you if you look at predicate logic essentially means that there is something in the internal structure which somehow has to be bring into a or take into a account classes relationships between classes of objects and not any argument that goes beyond singular objects an individual objects requires a different kinds of treatments. So, linguistically is this question of looking at adjectives which are like properties relationships classes of objects the relationship of individual classes essentially membership and classes and and this kinds of things somehow come out from the internal structure of the sentences and till you have actually your formula is then it is unlikely that they have going to be prove such a simple argument which is have; obviously, valued is also valid. So, there are voter kinds of arguments which we should be able to do with. So, like.

All men all cats are mammals all mammals are vertebra therefore, all cats are vertebra. So, those are those are like what known as categorical propositions the keywords in all the sentences all and is categorical in that since suppose think suppose think you had all cats are mammals some mammals do not have vertebra some cats do not have vertebra therefore, some mammals do not have vertebra for example,. So, so there are. So, there are some specializations af all and some and which make this categorical propositions. So,you need to look deeper into their structure. So, that. So, and mathematically speaking of course, actually its most obvious thing to do is to is to actually parameterize propositions is sought of parameterize propositions.

(Refer Slide Time: 07:59)



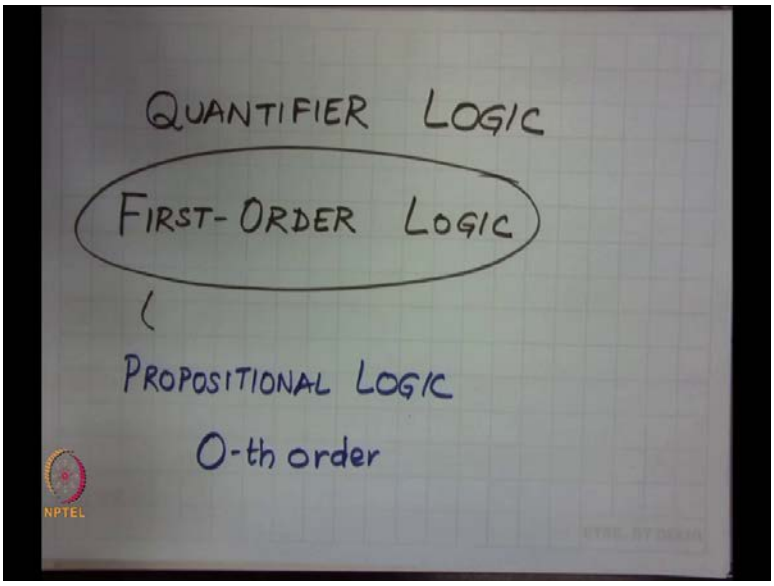
So, will have our parameterized propositions and these parameterized propositions are called predicates from which the logic gets its name. So, a proposition by whatever we have. So, far seen is something is a declarative statement.

For which at least theoretical there exists a truth or false will assign it a parameterized propositions is one whose truth or false cannot be establish from the proposition it self, but only when the parameters are replaced by singular individuals can you actually establish a truth or false conversations right. So, you take a property like. So, from elementary number theory you take a property like being number being a prime right prime x is a parameterized proposition prime x by itself where x is just a parameter of a place holder is not capable of being assigned a truth value true or false it can be assigned a truth value only if you replace that x by some individual number. So, like prime two that can be true or false. So, prime two becomes a proposition prime x is a predicate in the sense the it is just a parameterized proposition and the parameters are just place holders for replacing individuals for replacing the place holders by individuals right in which individuals can be substituted. So, a predicate does not become a proposition till all the place holders get plugged in ya and that can be more than one place holder. So, usually these predicates therefore, and by using predicates by using parameters essentially what you manage to do is that you manage also specify. So, you can think of this parameter parameter propositions you can also think them as like going from.

Propositions which are like constants two propositional functions propositions of somewhere you does. So, as a predicate you can also be thought of as a propositional functions. So, where the placeholders there might be many place holders and they establish a certain structure and truth or false would cannot be determined till values are actually substituted in the places in the various places right ya.

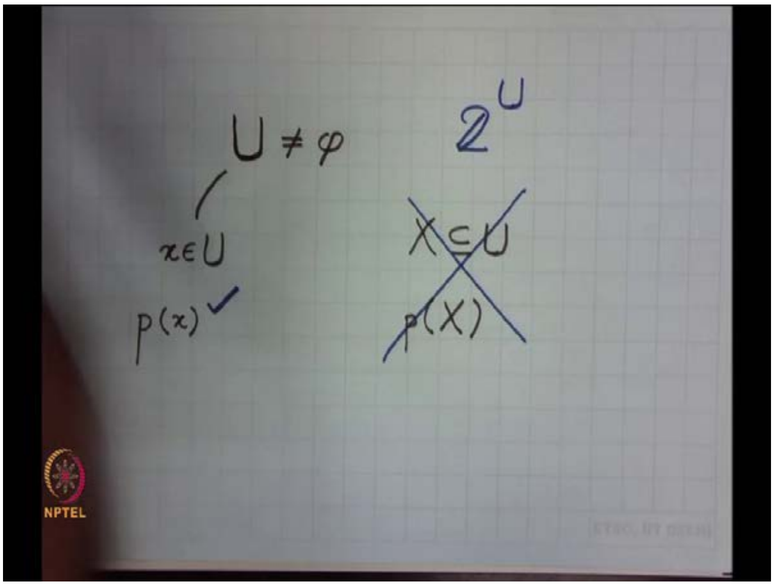
So, the other ways to look at things is that propositions of propositional logic are zero or a predicates and important thing here now is that we have this. So, essentially primary perpose beside linguistic besides a purely linguistic logic is also to that of describing mathematical theories right. So, when you look at any mathematical theory good theory for example, then what you have is a high prevalence of functions and relations and expressions involving functions and relations. So, which means that and they are about very often they are about these type of functions and relations often are used to describe indicatory objects or infinite classes in some fumitory representations right. So, bringing down the infinite to some finite impressibility is a proper exercises and mathematics that is something that we have to take into account and by or large.

(Refer Slide Time: 12:41)



So, the by and large whatever reasoning we do in this first order logic is more or less applicable to most mathematical theories there is settle instance there is stability about logic be in first order and what that actually means. So, what will do is lets look at lets think of propositional logic as it essentially being zeroth order. So, will say that propositional logic is essentially zeroth order ya when we talk about predicates or first order logic then what we are talking about is a class of individuals.

(Refer Slide Time: 13:40)

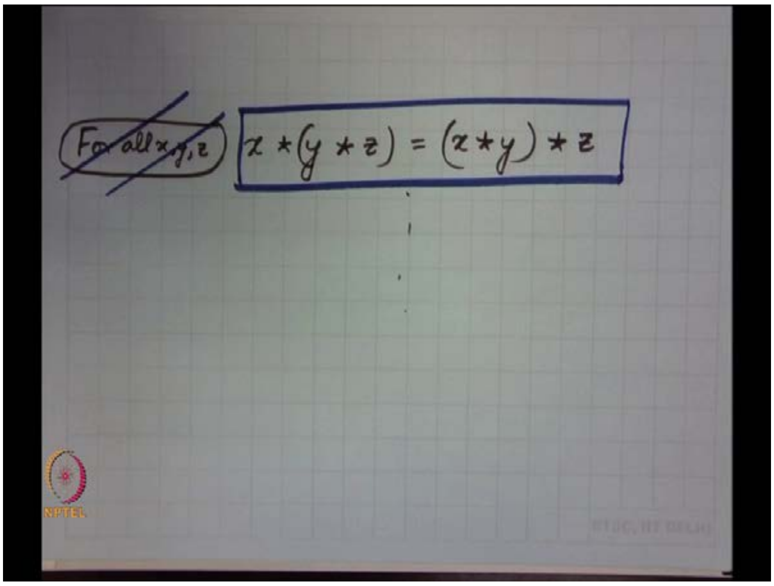


So, usually we are talking about the mathematical theory which has some universal set and this universal set is nonempty and what we are looking at properties of lets say elements of this universals set it is possible to look at p[properties of subsets of this universal set. So, what makes a logic first order is the fact that you are not interested in properties of subsets you are interested only in properties of individuals. So, you are looking at looking at fairly general properties about individuals only in that set and you might be able to class various characterize various subsets of this universal set by describing properties of individuals in some parameterized passion, but properties of classes or which essentially ab properties of subsets and being able to describe properties of classes of subsets of this universal set outside the domain that is at the movement. So, we are only looking at this we not looking. So, we are looking at essentially propositions propositional functions may be of the form p of x, but given subsets of u we are not looking at propositional functions of this I mean that is what this is outside the domain of this first order. So, we are only looking at propositional functions expressed in terms of individuals in the set and given that given that there is a subset of u which might for which this property p might be true we might be able to characterize subsets of u that way, but we will not be able to characterize properties of subsets of subsets I mean classes sub classes of this set right. So, we may not be taking into account this the movement you allow this also into your logic then what you are doing is you are getting into second order. So, essentially what then you are domain of this course is really is then two rise to u and then you have two categories of individuals little x and big x and then you can describe properties of this big x also. So, when you allow the description of that then also along with the description of that propositional functions on individuals you get second order the movement you go into two rise to two rise to u you get third order and fourth order and. So, and. So, ideally speaking the the mathematics does use higher order also many times. So, second order third order fourth order for example, various characteristic properties of topological spaces are expressed essentially they would essentially not fall in the domain of first order reasoning if you were to even the principle of mathematical induction is not really of first order property for example, it is a higher order property and. So, induction inductive principles are generally in higher order, but what, but ones you have studied first order atleast the logical extensions to higher order are sought of obvious the same kind of the same kinds of quantifiers quantification rules and. So, on apply and therefore, you do not. So, its possible to actually look upon first order logic essentially being officiate to describe most of mathematics right. So, that is that is what we are going to do.

No no what you are saying is you are not saying you are not saying element by element you are talking about in induction what you are saying is a property p of elements and therefore, your inductive your induction is saying for all properties p of individuals if lets say p is true of zero and the fact that p is true of n employee p is true of n plus one then p is true for all the naturals lets say, but your induction principle actually says this is true for all properties p right. So, in that sense you are actually quantifying over the set of properties of natural numbers and you take any property of natural numbers that characterizes the subset of the naturals nad when you are saying that you are making the all properties of natural numbers you are saying all such types of naturals and. So, your induction principle the principle of mathematical induction you are quantifying over the all properties of over individuals in that sense it is atleast second order what one can do of course, is one can restrict as its to the particular order and say you take the principle of mathematical induction restricted to first order so; that means, you are just looking at properties of individuals and you just characterize may be subsets of naturals. So, induction otherwise is actually had a property because implicitly induction quantifiers over all properties will satisfy this pattern.

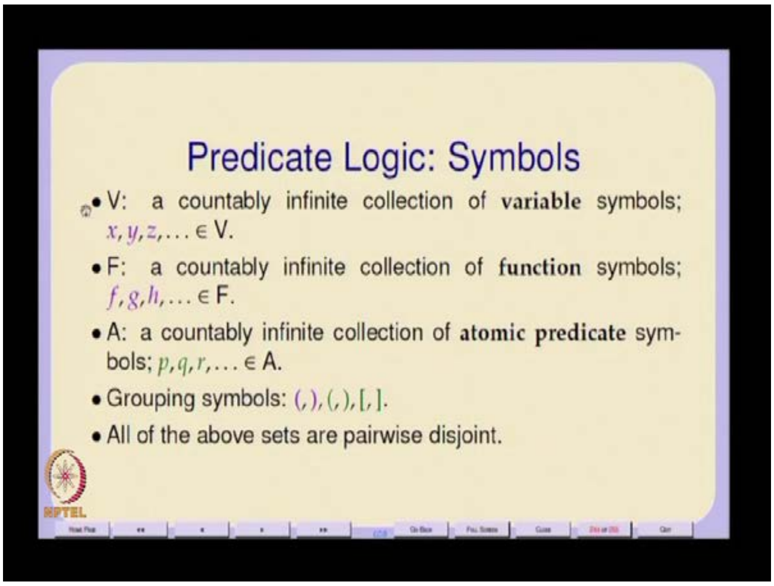
So, lets another problem in general with mathematics textbooks and mathematics teaching is that this exact nature of quantification is never made clear. So, for example.

(Refer Slide Time: 21:38)



If you look at if you look at say the group theoretic axioms I just I just say x y z equals x y z and. So, on and. So, fourth right, but there is actually a quantification for all individuals x y and z very often mathematicians just do not right this many algebraists will just write this in very much the same way in which we wrote the rules of inference we are looking at x y and z as particular places and you are looking at skeletal structures you are saying the this the skeletal tree of the left side transform to the skeletal tree on the right side for all possible substitutions also you are looking it as a macro or a skeletal structure a pattern matching structure rather than as a quantified statement. So, that this is a one thing that is one thing. So, in general what actually happens is that when a statement is valid then often it is not nessasary to quantify it. So, that is one thing and. Secondly, for the purpose of validity what happens is that if this is valid then its not necessary to quantify it. In fact, what they can show is that this is valid if and only if the quantified statement is also valid. So, let us therefore, mathematicians often takes these shortcuts what happens is it can be confusing for a beginners, but usually it is a logically valid in a strict logical sense sure.

(Refer Slide Time: 23:57)



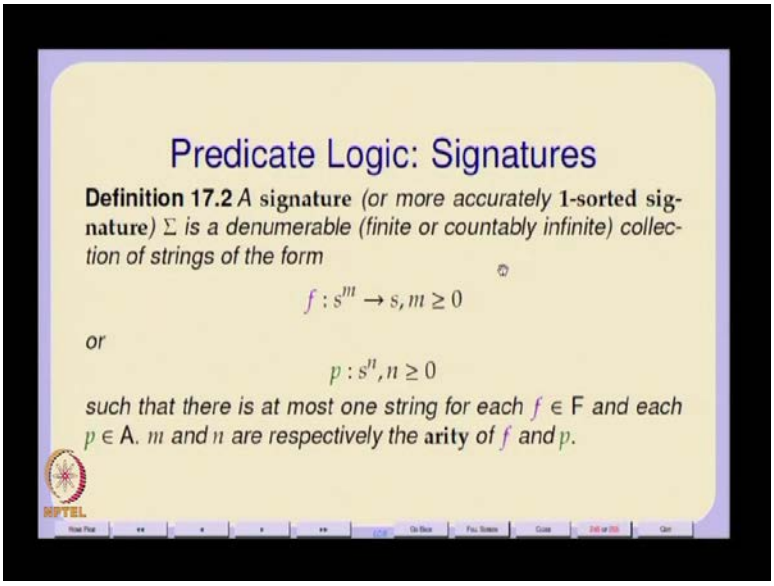
So, what we will do we will start with the syntax of predicate logic. So, I have v of count ably infinite set of variables. So, as we said we are talking about propositional functions very much like the way we move from lets say constants in very much like the way we move from arithmetic to algebra just look at your transitions in your education from arithmetic to algebra in the case of arithmetic you had specific constants and you delt only with specific constants.

When you move to algebra you actually graduated to using placeholders without actually looking at specific constants. So, the introduction of algebra actually gave rise to the notion of variables. So, so. So, when we graduate from propositions to predicates which has propositional functions we are actually graduating from constants to variables. So, you require placeholders. So, this ask you count ably an infinite set of variables and we also seen abounded infinitely collections of function symbols in any mathematical theory functions play a very important role in that operations and functions will regard them and then there are in additions to functions of any mathematical theory also has collections of relations and those relations are what we going to specify by predicates. So, we have an infinite collections of atomic predicate symbols right then we use these grouping symbols.

Is the color distinction is clear much clearer than here it is here.

So, so. So, will use this grouping symbols I have a peculiar problem that scopes of unary operators and bindings is always is something confusing. So, I am using this square brackets most logic books will not have the square brackets and of course, all these about sets are disjoint are separate sets right.

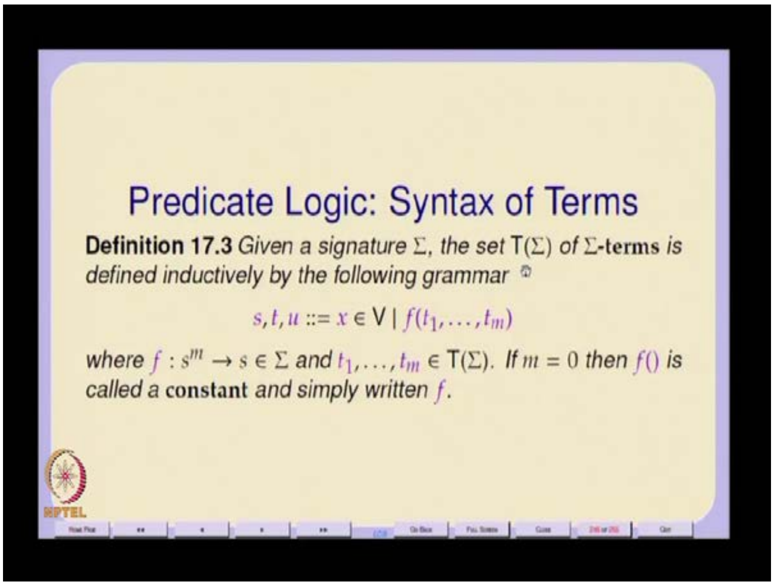
(Refer Slide Time: 26:20)



And will start an algebraic version and the reason is actually historical and philosophical if you look at I mean historically speaking if you look at development of mathematics from from India for example,. So, the development of mathematics is essentially algebraic nature started with numbers the place value systems arithmetic and then went on to expressing various geometrical aspects for example, the ((_)) theorem purely algebraic as an educational form as an identity and then trigonometric relations and. So, on as essentially as in an arithmetical or in an algebraic form you can any kind of algebraic form is also it also has a certain syntax and therefore, you can think of it as a linguistic form. So, what actually pervaded most of ancient Indian mathematics is high propotion of algebraic reasoning. So, and therefore, linguistic reasoning also. So, if you look at formalization of transcript by Panini that is essentially like defining the formal grammer rules very much in an algebraic sense right on the other hand if you look at the development in the Greek essentially starting from Egypt band then moving on to Greece and then coming back to Alexandria at classify all of them are essentially Greek mathematics the problem was there was not of numbers symbols and manipulation the primary problems was there of a geometric nature being able to predict when the nile is going to flood the plains being able to predict the course of the nile river for example, being able to do triangulation being able to walk across the desert some navigational aid using the stars. So, on and. So, forth using the sun in some geometrical fashion the angle of the sun and. So, on the entire development of mathematics actually was divorced from they were numbers of course,, but it was primarily geometric in nature angles lines straight lines plains heights distances. So, on and. So, forth and even up to the eleventh and twelfth if you look at the problems of Greek mathematics including the which we see in the algebraic most of their solutions for example, how do you solve quadratic or cubic equations most of those expressed quadratic expression terms of areas and cubics were expressed in terms of volumes. So, what we would regard as a simple cubic equation to be solved the Greeks would actually be imagine volume of rectangle cuboids and cubes placed in certain fashion or removing sub cuboids and cubes from other cuboids and cubes. So, subtraction is essentially removal of volume and then resulting volume and So, on and so, forth even the solution of expression had an absolutely geometric equations and it was always they always do diagrams in some kind of respective mode some kind of isometric mode and actually try to reason using volumes and areas and they had to deal with second degree equations and third degree equations so, but the whole point was that the actually the whole of geometry called algebra ones you have a coordinate system and a notion of a point is an ordered pair of real numbers. So, on and. So, forth and algebraic properties of real numbers your algebra is geometry and therefore, in certain sense the interaction algebra and geometry. So, so if you take the development of differential and integral calculus this is also an algebraization of course, and. So, algebraization is part of linguistics and therefore, if you are looking at the formal language development we look at all mathematical theories in some algebra is formed right we look at signatures we look at any mathematical theory will think of it as contain a signature. So, this signature essentially gives you a finite or infinite collection of function symbols with their rarities essential and specification of there with their rarities and these predicate symbols have their rarities. So, s s is just a formal symbol for a sort. So, there is a sort. So, what we are looking at are one sorted signature if you were to look at a typical programming language a typical programming language think of as a many sorted algebra several sorts like Booleans naturals integers reals strings and charecters may be and on this there are operations between sorts instead of using a single symbol s we will use many symbols s one s two s three to specify essentially the signature of an algebraic system and this notion of a signature is exactly h what your modern function programming languages are also adopt hat is they have this notion of a sort. So, for example, animal or camel or Haskell has these int and bool and char and. So, on and. So, forth. So, they have this notion of a sort animal or camel or Haskell has this int and bool and char and. So, on and. So, forth signature essentially in terms of functions and predicates these function symbols and predicate symbols while specifying essentially there rarity yes right and the sort of the domain the code in the case of functions in the acse of predicates of course, they are thought of programming languages as functions which take you to bool. So, that is the only functions. So, we look at these. So, will assume got some signature specified each function symbol and each predicate symbol and will assume there is no conflicts every function symbol there is a unique string specification in terms of the sort and we rae looking at one sorted signature which essentially means in mathematical description we are looking at this we are looking at this u as a single sort the generalization to many sort signature is very hard.

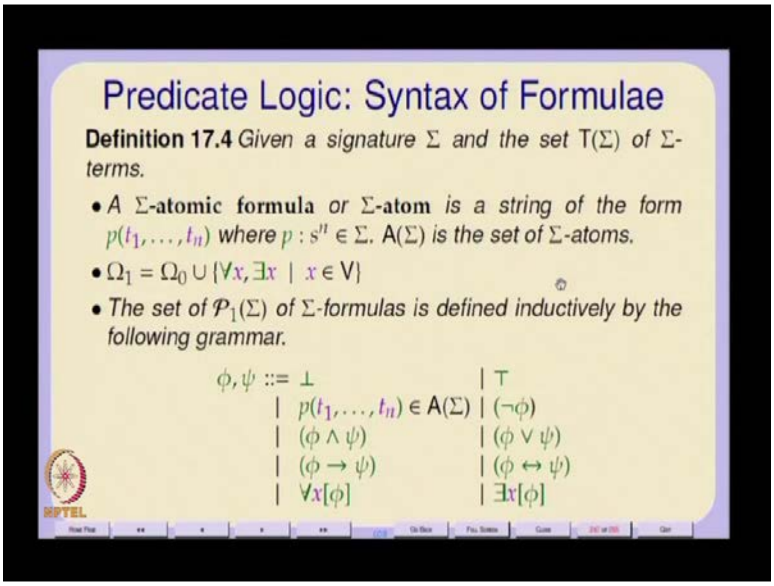
One could do that too, but it becomes more complicated and tedious, but there are no technical problems in the generalization, but there is also another thing that it is possible to take the many sorts that you might have put them all together and call them a single universe and have distinguishing predicates for the membership in the various sorts. So, we can have membership predicates essentially for the to distinguish between various predicates if you had a collection of disjoints then we use one predicate to essentially use one predicate in a sort right. So, then net effect is that either case we have many sorted algebra becomes tediously, but even the many sorted algebra given a finite number of sorts can be described through one sorted algebra adding this extra one membership predicates in the signature right. So, for example, the simplest thing is to take integers and booleans put them together take a disjoint union of them and just have one predicate we specifies whether some arbitrary object they can think is either boolean or integer and have these two predicates integer something and boolean something and you can actually every thing that you can do with many sorted atleast a finite number of sorts you can do in a one sorted case.

(Refer Slide Time: 36:22)



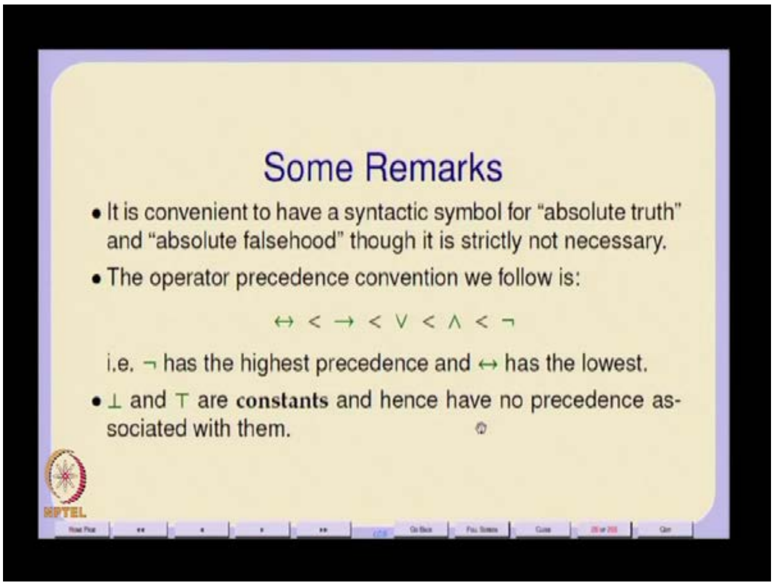
So, that is important write now and of course, we have this notion we have this we have we essentially define a term algebra. So, given a signature sigma and we will just assume a one sorted signature we have the set of terms sigma terms define inductively by this grammar and I will use this violet color essentially to terms. So, if f as a signature if f m arrow s then f can essentially take other terms parameter it is an inductive is an inductive is an inductive definition I have gone beyond the usual b n f grammar by specifying also other things in short hand s t u are like artificial typical members of terms. So, I will use this symbols like s t and u in violet for to denote terms. So, this x belongs to v actually says that all the variables are terms and. So, of course, in particular case m is zero means that you are a constant and instead of writing it as f these powerful parenthesis. So, that is a accent tags of terms and I have given them separate color because actually specify tour mathematical theorem except for the relations that are there in the theory for which you are going to have predicates.

(Refer Slide Time: 38:02)



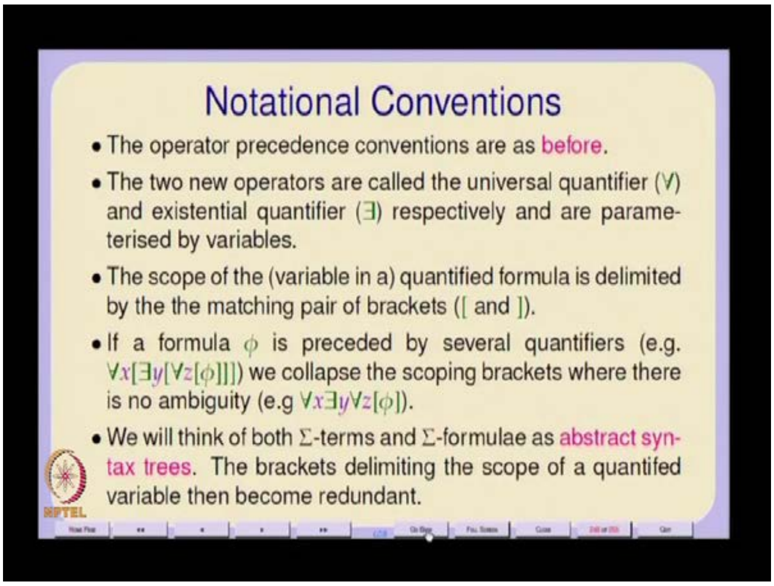
So, given a language of terms already sigma terms then you also have a collection of predicate symbol atomic predicate symbols this p q r etcetera for atomic predicates symbols are essentially propositions functions si they take they have predicate functions empty places to be filled up. So, an atomic propositions itself means that it will have a propositional form in which this placeholders can be replaced by terms t one to t n and then will have all the propositional connectives after all we cannot I mean you are going to consider compound propositional form still. So, we will have the propositional connectives that we had I am now calling that set omega not to signify that they are all the propositional connectives like not and and and or and conditional and bi conditional and then in addition we have today operators called the quantifiers if you have an infinite collection of variables you; obviously, have an infinite number of operators here one for each variable one this is for all one for all for each variable and there exists each variable. So, we have this is are signature these are collection of operators and we will use five size elements as typical members of the at of formulas and the language p one of sigma. So, propositional logic I just called at sigma now since it is parameterized and it is made to describe some particular sigma algebra we parameterize it. So, p one is a first order language for describing for describing the essential properties of sigma algebra. So, this sigma formulas are defined in inductively this way and here is my peculiarity that is that I specify the scope of variables or quantified variable and I like to deal it them by square brackets I personally have lot of trouble I just remembering residences of operators and figure out quantified formula its really painful instead had a clear delimitation and scope they have in programming languages now I do not need to talk about the precedence of operators and symbols what is the extension of this scope is clearly delimited by the brackets though that is an extra piece of syntax.

(Refer Slide Time: 41:42)



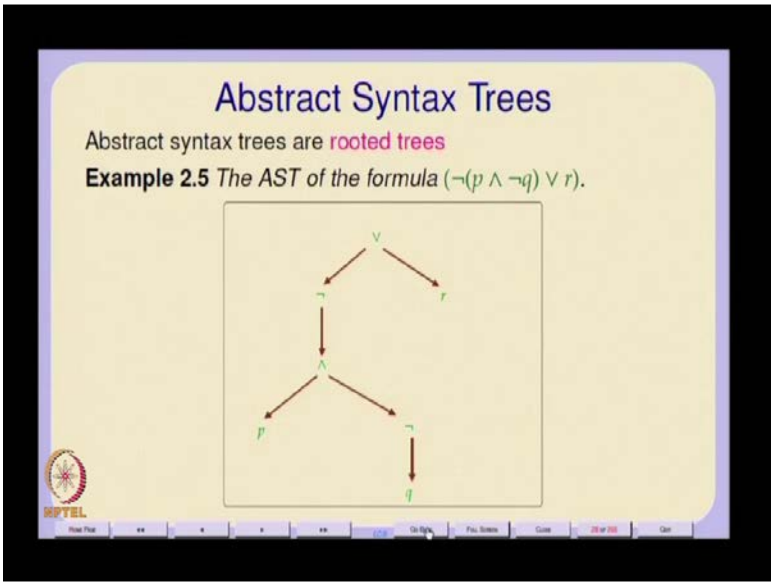
So, the operator precedence as before for these new operators particular because anyway the square bracket should take care of it; however, what I am also going to do is that I am going to have a short cut having too many brackets is also a terrible.

(Refer Slide Time: 41:50)



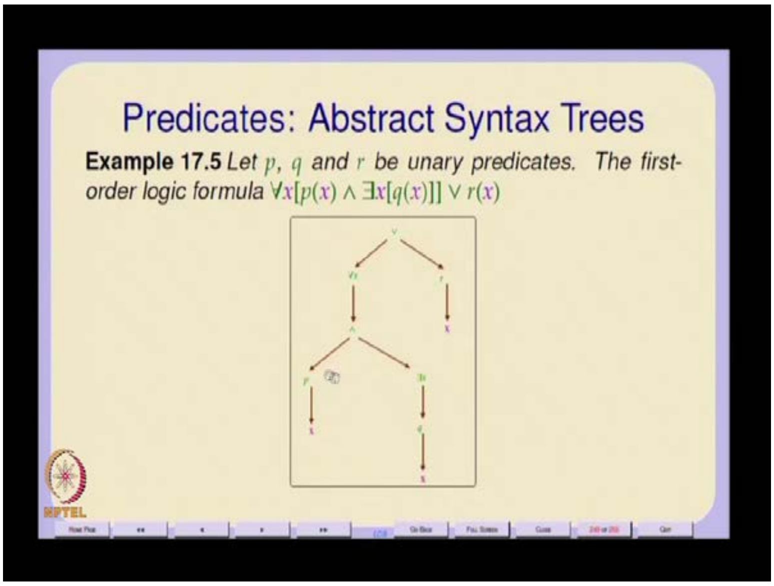
So, when you have a formula five proceeded by several quantifiers as in sequence no other propositions are occurring in between sequence of quantifiers then use a single pair of brackets. So, instead of here you look at this actually exists for y which has three pairs of square brackets to specify the scope, but it is simpler to just specify that for all x exists with a single pair of brackets are undefined this essentially tells you the scope you do not lose any information and it is simpler to write that is what we will do. So, and of course, as before we are not really interested in like most logic in strings and distinguishing between strings interested only in the abstract syntaxes of these formula.

(Refer Slide Time: 43:03)



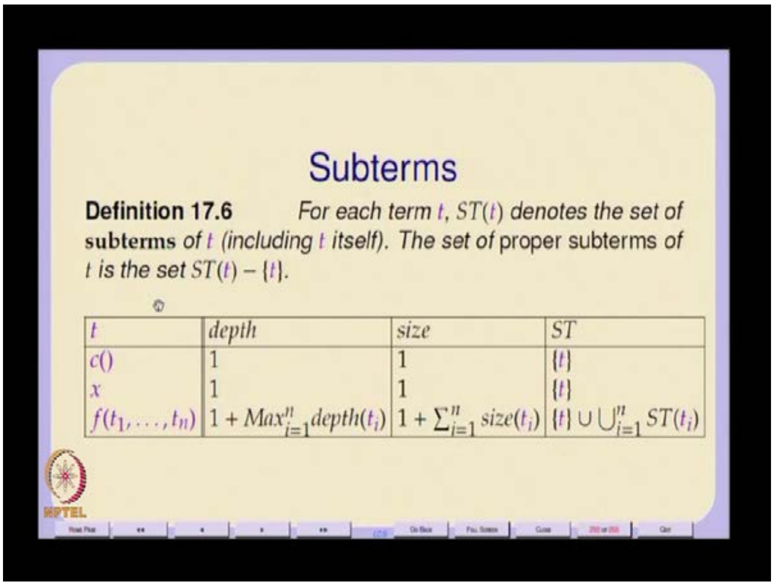
Any formula can give you an abstract syntaxes here is some here is some this is a propositional formula they are the abstract syntax tree and in an abstract syntax tree even this scoping brackets become redundant I mean you need them because scope also follows the tree structure. So, it is actually. So, we think of them abstract syntax trees right.

(Refer Slide Time: 43:32)



So, you can see all the operators here it is too small, but here is the formula is like for all x there exists a x and q x or r x this is abstract syntax trees for that the root operator is this or and then r is. So, abstract syntax trees starts with predicate symbols and then becomes terms and there also abstract syntax trees. So, for example, I can have and there exists q x and here is the this and p of x and this or is then there is universal quantifier for all x and this is the abstract syntax tree of this formula if instead of x I had some complicated term then that could have here for example, in essence. So, we have green labelled nodes which gradually go into violet labeled nodes or abstract syntax trees look something.

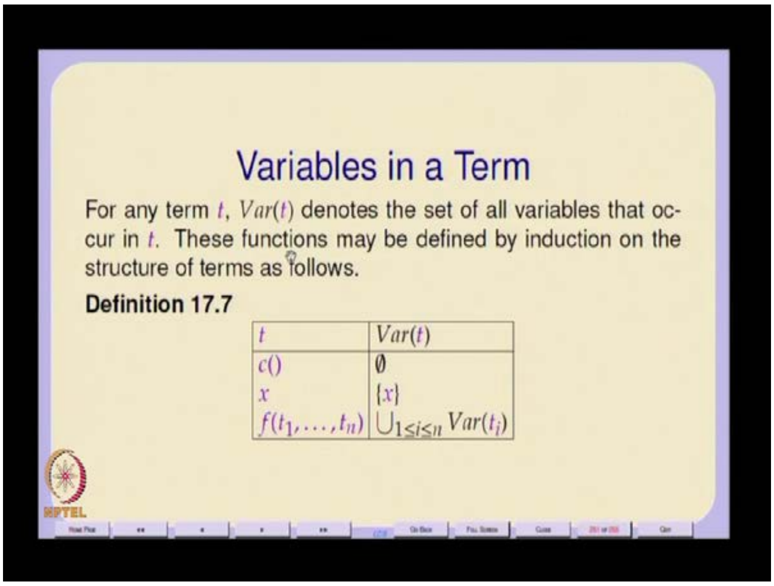
(Refer Slide Time: 44:49)



t	depth	size	ST
$c()$	1	1	$\{t\}$
x	1	1	$\{t\}$
$f(t_1, \ldots, t_n)$	$1 + Max_{i=1}^{n} depth(t_i)$	$1 + \sum_{i=1}^{n} size(t_i)$	$\{t\} \cup \bigcup_{i=1}^{n} ST(t_i)$

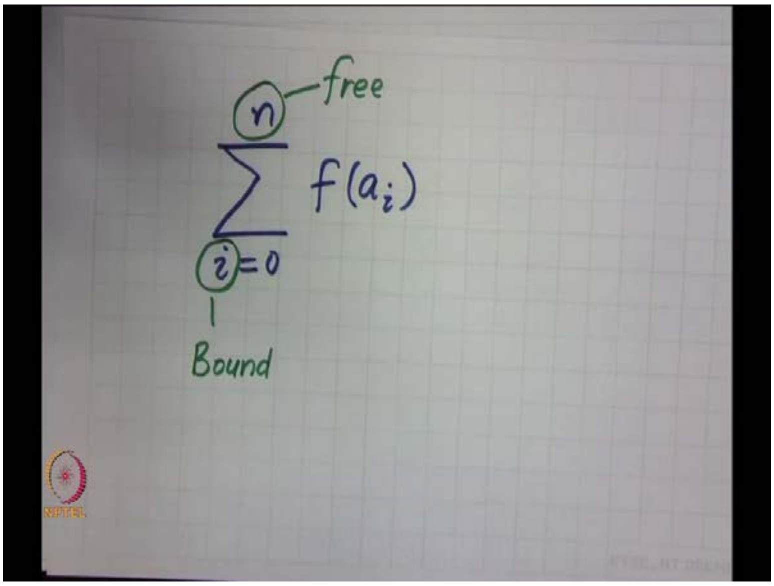
That is a far more amount of technical staff that we related to the syntax that we need to work about at the notion of now we had to right the notion about free variables and bound variables notions like sub terms and sub formula. So, we have a hierarchy now clearly we have a two level hierarchy is the hierarchy of the language of terms and above it there is a hierarchy of the language of predicates and you cannot mix the two all your abstract syntax trees should at the root look green they should either be completely popular if they are terms or they should start from somewhere green and start becoming perfect with in that purple they cannot change to green for example,. So, we have the hierarchy of two level hierarchy of predicates right. So, actually brings in notion of sub terms for a term and these are the measures that will be useful for performing induction like depth size of a term. So, on and so, forth which can be defined by the induction of the structure of the terms.

(Refer Slide Time: 46:11)



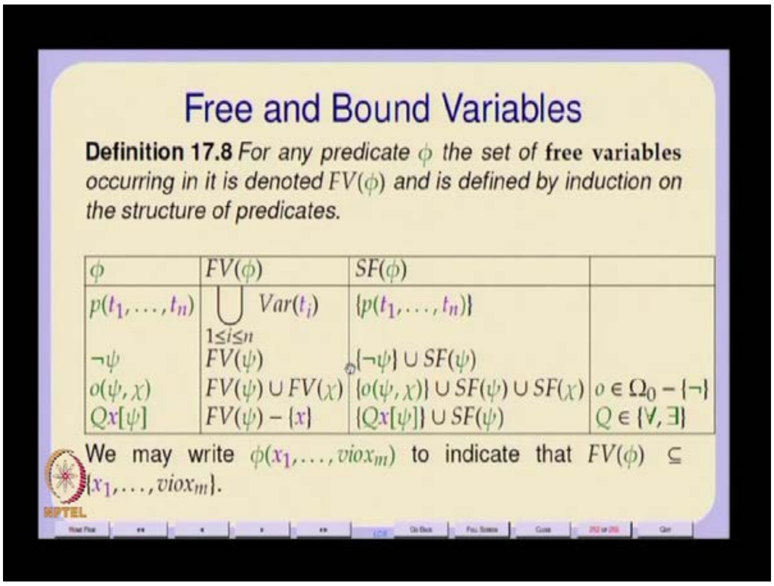
And similarly, but the other complication which quantification brings is the notion of a binding and scope and the notion of therefore, free and bound variables right. So, we will in any term of course, there is something there is another simplification you have done remember that our language of terms did not have a bound variables whereas, strictly speaking.

(Refer Slide Time: 46:44)



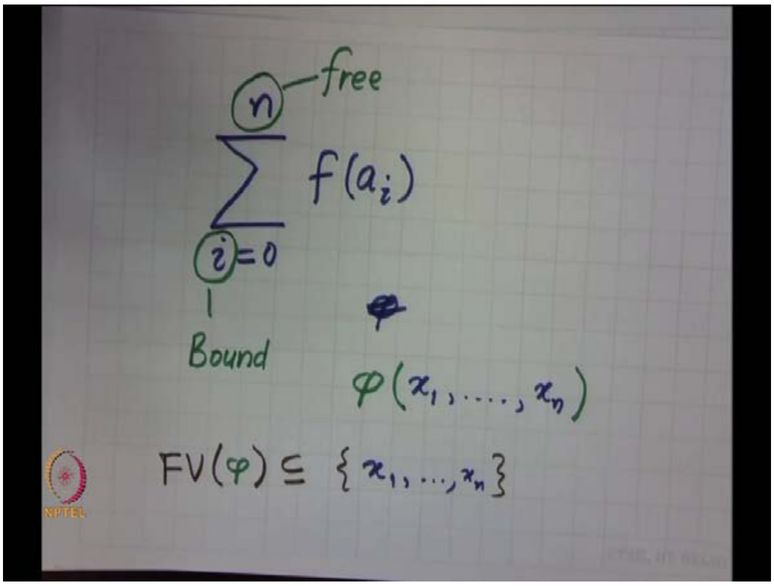
If you have when simple things like this then the notion of free and boundary actually have to come up because even in the language of terms, but we are simplifying for example, this is a bound variable where as this is tree right I mean. So, we are not our signature specified in such a way that there are no binding operations and therefore, there is an no concept of a bound variable in terms that is a simple verification we have done, but we will just look at the variables in a term that can be defined by induction on the structure of the terms.

(Refer Slide Time: 47:41)



You need the variables in a term in order to able to be defined free and bound variables on predicates right because your terms form the underling subtrees of a predicates. So, we will define for any predicate fv of five set of all variables of course, given an atomic predicate p of tone to t n where one to t n are terms you just take the union of all the variables in that in each in all the terms and that is the at of free variables and we also sometime require the notion of the free variables formula, but notion of a sub formula is quite is just that of a sub tree of that except that you should not look at sub trees terms as subformulas I mean the violet the sub formula should have something green at the top they cannot just be violet. So, so in the case of not psi you just have free variables of psi set of formulas is just this for any of the propositional connectives except not you just take the union of the free variables of the free components all of them are binary operatories and similarly you also define your sub formulas as unions like this and for any of the quantifiers you remove the quantified variables from the set of free variables of the body of the quantifiers. So, this whatever within the square brackets and the scoping brackets is called the body of the quantified formula and x no longer remains in that because x is a bound there right I am sorry this should have been changed to I forgot to put a backslash. So, x one to x m all in violet color actually some times what we will do is we will just right a formula in this passion.

(Refer Slide Time: 50:18)



So, we will write this to indicate that the set of free variables five the set of free variables of five is contained in the set I just stop with this is contained in this set; that means, is not necessary that all of these variables are used some may be used. So, we will we will continue this in later and will stop here.