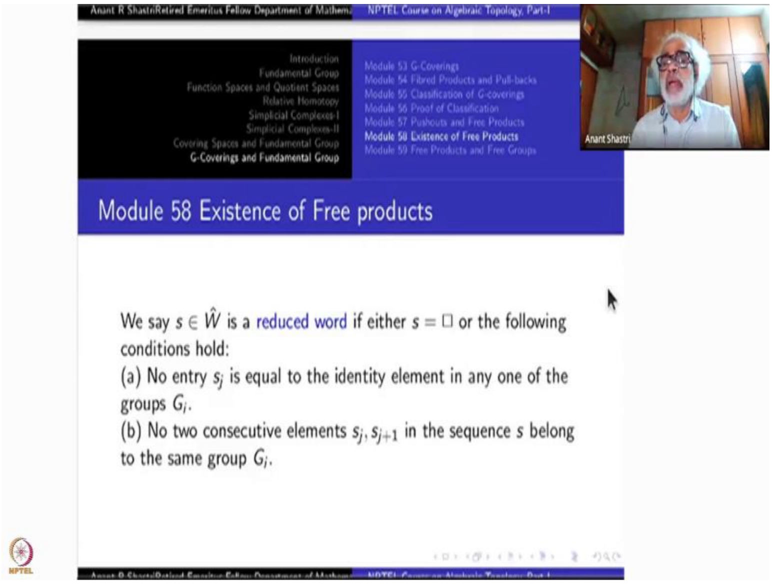
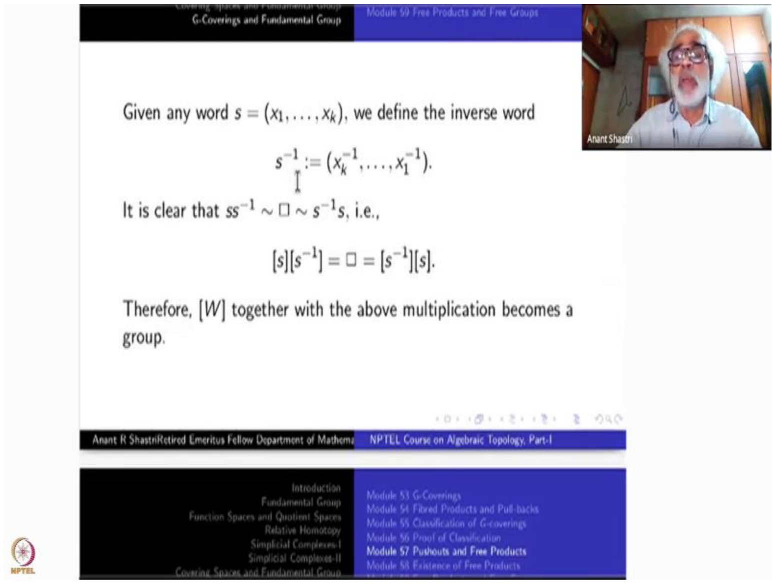
Introduction to Algebraic Topology (Part- I)
Professor. Anant R. Shastri
Indian Institute of Technology, Bombay
Lecture 58
Existence of Free Products, Pushouts

(Refer Slide Time: 00:17)



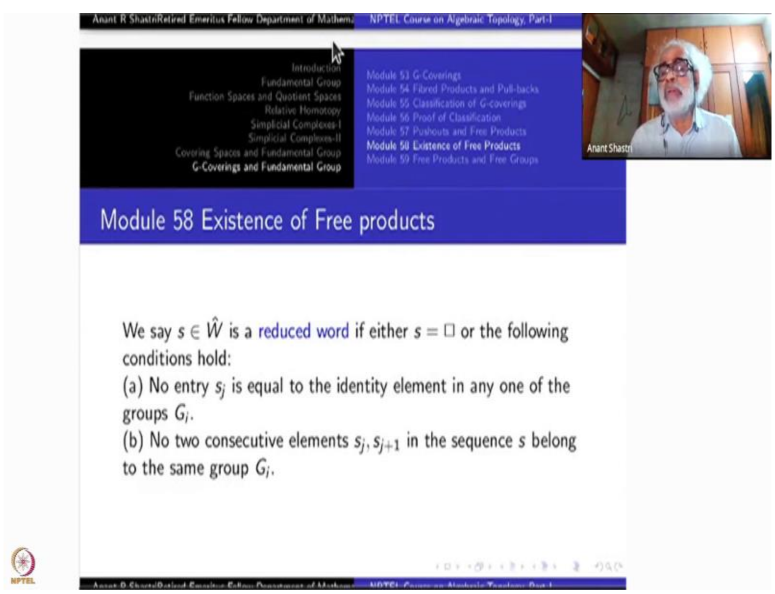
Last time starting with the family of groups, we took the disjoint union of these groups as our alphabets. Over these alphabets, we reconstructed what are known as free words which are nothing but finite sequences of alphabets. On the free words, we introduced an equivalence relation and that made it into a group namely, with the binary operation of `concatenation of two sequences along with the empty word as the identity element. The set of sequences itself is a semi group, associative binary operation with a two sided identity. But it is not a group. To make it a group, we took equivalence classes of sequences, namely equivalence relation was defined by a chain of elementary collapsing or its inverse finite chain of such a thing, if two words are related then they are said to be equivalent.

(Refer Slide Time: 01:51)



So, these equivalence classes were denoted by [W]. This set of equivalence classes becomes a group under the obvious inverse operation namely take a sequence $w = x_1 x_2 \cdots x_n$, take inverse formerly as inverses of each x_i written in the opposite order, $x_n^{-1} \cdots x_2^{-1} x_1^{-1}$. These inverses make sense because they are elements of various groups that you have taken, this on the right hand side these universes are genuine inverses of group elements on the left hand side this inverse is a notation. Now, we can verify that when you go to the classes, equivalence classes that inverse is equal to empty set therefore, on the classes this is the inverse. So, this much we had seen.

(Refer Slide Time: 02:50)

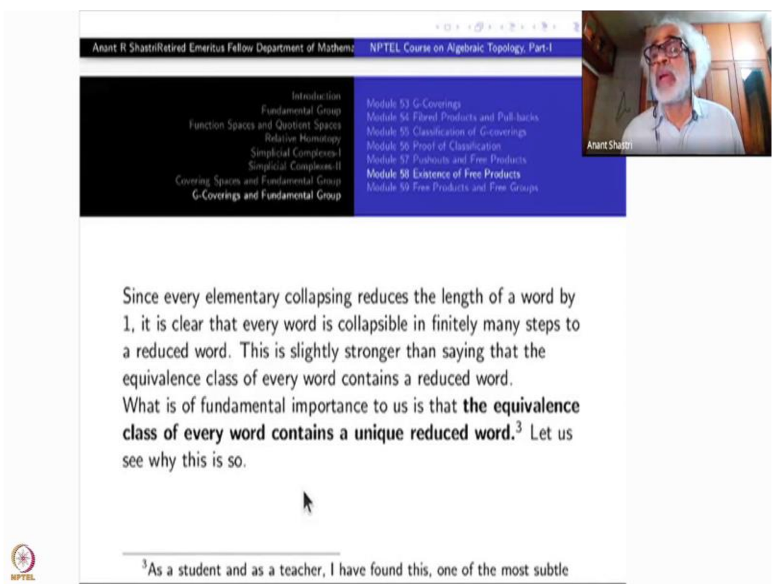


So, today we shall complete the proof of the existence by showing that this bracket [W] is the one which satisfies the required properties namely the universal property. So, we introduced the terminology a every word is called reduced word if either it is empty word is reduced word or the following conditions hold, none of the entries of this free word should be an identity in any of these groups that is the first condition.

The second condition is no two consecutive elements should belong to the same group. Remember, under these two conditions, we can collapse whenever such a condition is there, whenever any element is identity element we will delete it. So, that is the first collapse whenever two elements are consecutive and belong to the same group, again we call out by writing them together as one single element in the corresponding group.

Therefore, these two operations if performed finitely many times on any given word will produce a redcued word equivalent to the given word. After all we start with a finite sequence. So, after finitely many steps, no more collapsing is possible because each time you perform this the length is reduced. Therefore, what we have just observed is that in the equivalence class of every free word there is always a reduced word.

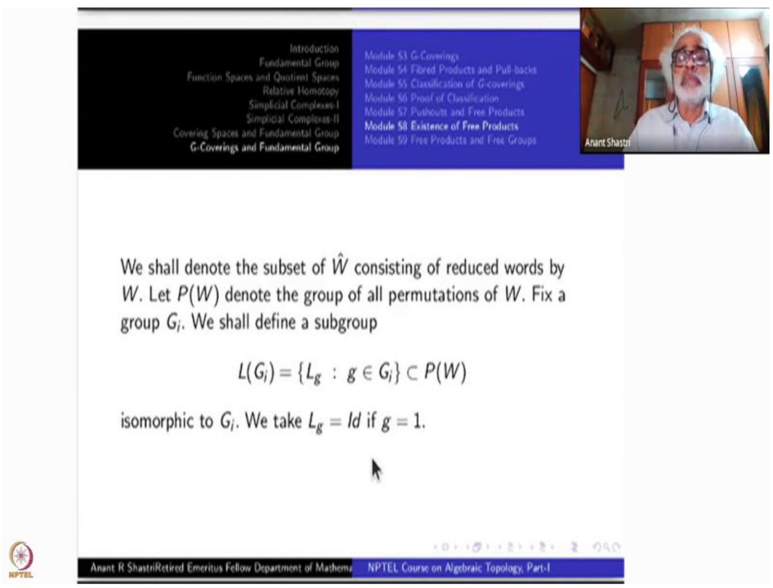
(Refer Slide Time: 05:11)



I mean it can be empty also, empty word, by definition reduced. So, the point is that if I say there is always a reduced word in eahc class, it is weaker than saying that every free word in finitely many steps can be reduced to a reduced word i.e., can be collapsed to a reduced word. --actually `collapsing' nothing but finite many collapsing collapsing …. collapsing. That remains in the same equivalence class. What is not clear is that suppose you have a very huge word collapsing can take place in different ways and different directions, different order. Then it is not at all clear that you will get the same reduced word. Two people, or two different kinds of algorithms are written down for reducing, starting with any free word, whether you will get the same reduced word or whether there may be two different reduced words representing the same class, this is not clear.

So, what is of fundamental importance to us is that the equivalence class of every word contains a unique reduced word. This what we want to see and this is more or less the key to the universal property of this the group that you have constructed. They go hand in hand.

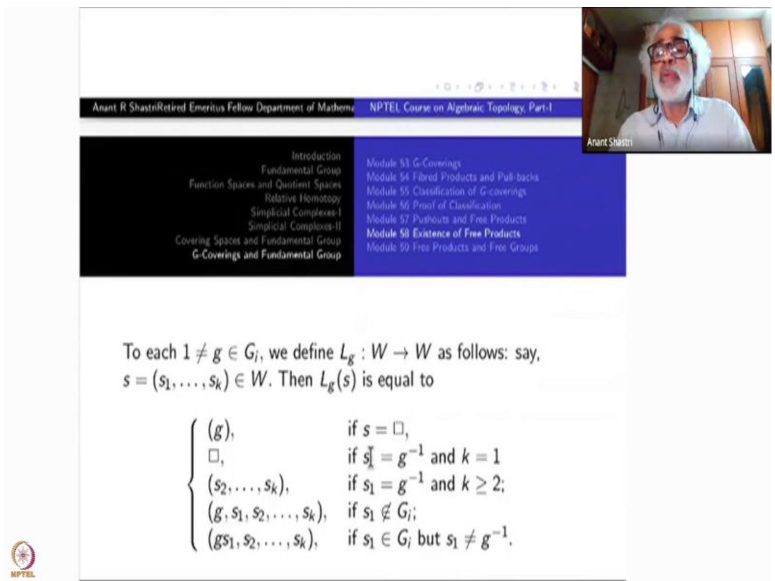
(Refer Slide Time: 07:05)



So, study this one carefully. So, here is a technically superior method. This method should be learned it can be useful elsewhere also. It is actually used in linear algebra quite often, linear algebra and functional analysis and all sorts of algbra. What we do is, we shall denote the subset of $\hat{W}$ consisting of all reduced words by removing that hat i.e. W. So that is the set of reduced words. Also the set of equivalnce classed of reduced words will be denoted by [W]. So, what we would like to show is that [W] is in bijection with W. So, for seeing this one, what we do is take the permutation group of W, W is now reduced words not the whole of $\hat{W}$. The permutation group is a very huge group of course. That is the key. Let this $P(W)$ denote the group of all permutations of W. Fix one of the groups G_i; concentrate on one of the G_i. We shall define a subgroup $L(G_i)$ of this $P(W)$. I will denote, for each $g \in G_i$ an element $L_g \in P(W)$ which we are going to define and then take $L(G_i) = \{L_g \ : \ g \in G_i\} \subset P(W)$. The correspondence $g \mapsto L_g$ is going to be a monomorphism $G_i \to P(W)$ so that $L(G_i)$ is a subgroup of $P(W)$ isomorphic to G_i. This is the first starting point of this universal property though.

So, if $g = 1 \in G_i$ is the identity, take $L_g = Id_W$, identity permutation. After all, we want this assignment to be to be a homomorphism.

(Refer Slide Time: 09:47)

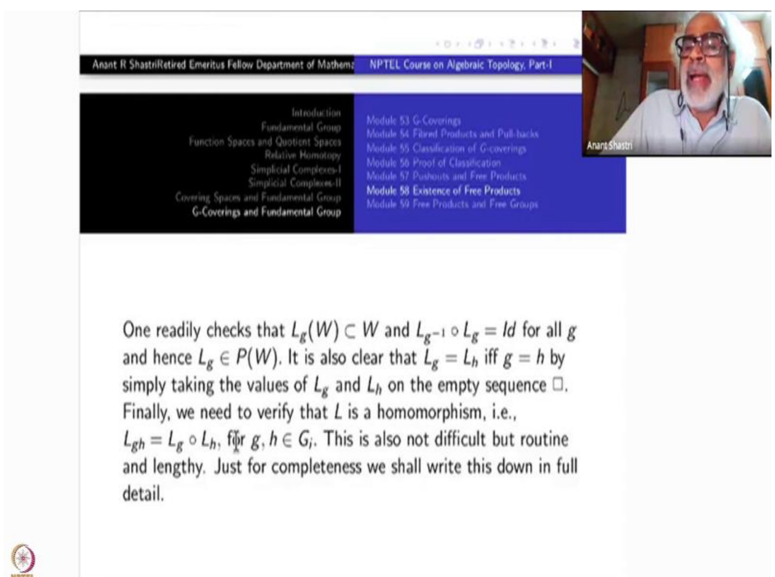


Let L_g be defined as follows. So, we have to have patients because we want L_g to take a reduced word to a reduced word. If you are dealing with $\hat{W}$, this would be easier. But here we want reduced words to remain reduced. The very notation L_g indicates that we are going to multiply on the left by g. So, what we are going to do, we are going to do a left multiplication on W, but then we have to reduce the word so obtained.

So, now take the case when g is a non trivial element of G_i. Let $s = (s_1, \ldots s_k), k \geq 0$ be a reduced word. Then $L_g(s)$ will be defined as follows. If s is empty then $L_g(s)$=(g), is just the sequence little g, namely one single element. If $k = 1$ and $s_1 = g^{-1}$, then take $L_g(s)$ to be the empty word. Because left multiplication we produce the sequence (gg^{-1}) which is not a reduced word so you perform a collapse so that you get the empty word. So, $L_g(g^{-1})$ is defined to be the empty word. Next suppose k is bigger than one (but $s_1 = g^{-1}$) then all that you have to do is to cut out the first two letters and write down the rest of the sequence $(s_2s_2 \cdots s_k)$. Remember that s was a reduced word and if you cut it short at either initial part or end part, it will be still a reduced word. (If you cut out somewhere middle portion then it may not be reduced.)

So, I have dropped out s1 here. This is still a reduced word. Now, up to these three case are over. The next case is s_1 is not an element of G_i at all. Then just put the g extra and $s_1s_2 \ldots s_k$ as it is. g will come on the left. So, $L_g(s) = (g, s_1, s_2, \ldots, s_k)$. Since $s_1 \notin G_i$ it folows that this is a reduced word. The last case is, suppose $s_1 \in G_i$ but it is not the inverse of g. Then you just combined g and s1 into one single letter $gs_1 \in G_i$ which is not identity. So, this gives a reduced word again. So, that is the complete definition of what is the meaning of Lg of s. Lg take a reduced word to another reduced word. Therefore, for each g, L_g is a function from W to W. We have to show that it is a bijection, then only, it is in $P(W)$.

(Refer Slide Time: 13:13)



Thus we have seen $L_g(W) \subset W$. The bijection part is taken care very easily by just checking that $L_g \circ L_{g^{-1}} = Id_W = L_{g^{-1}} \circ L_g$. Go through these steps in the definition one by one.

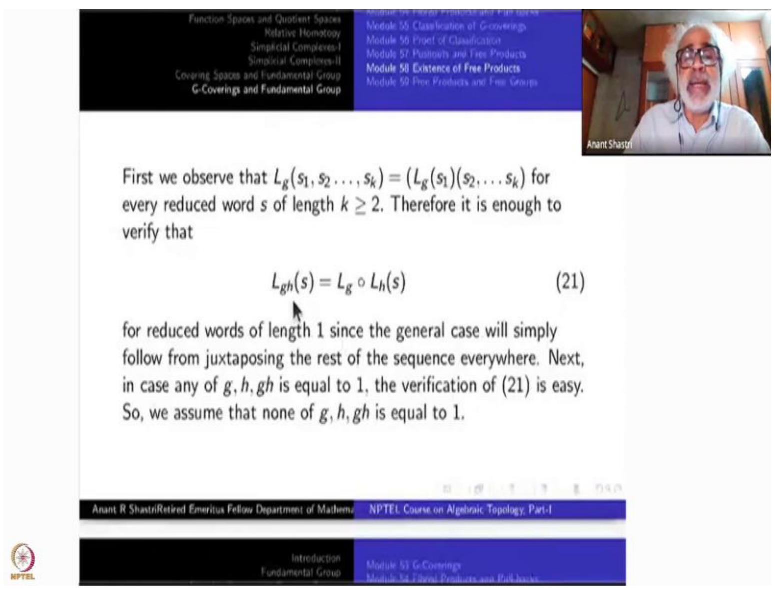
So, L_g is actually an element of $P(W)$. It is also clear that if $g, h \in G_i$ and $L_g = L_h$ then $g = h$ becuase you can operate both sides on the empty word. These are two permutations of W, L_g

opeartinh on the empty word is the reduced word (g). Therefore the map $g \mapsto L_g$ defines an injective function $L : G_i \to P(W)$.

Finally, we need to verify that L is a homomorphism. see L itself is injective. Each Lg is a permutation, that is different than saying L itself is injective and now I have to verify that L is homomorphism that is for all $g, h \in G_i, L_{gh} = L_g \circ L_h$. but you have to have patience and verify everything properly.

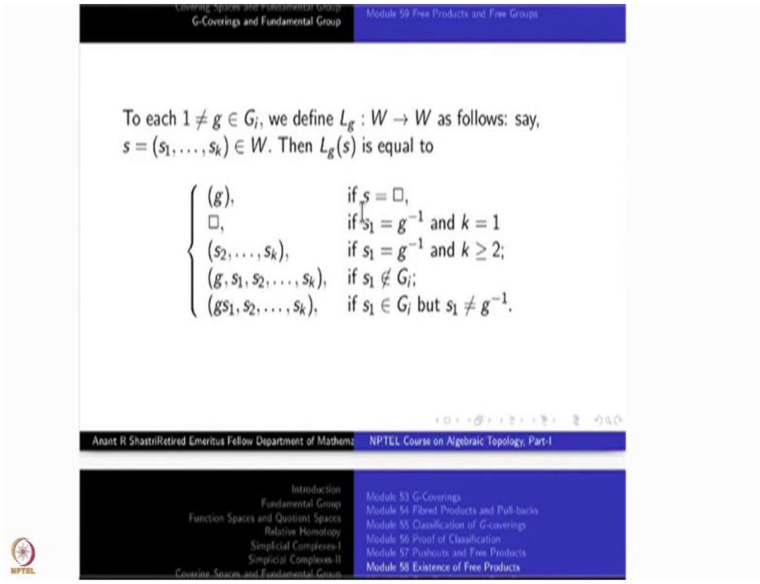
So, in order to show this what you have to show that $L_{gh}(s) = L_g(L_h(s))$, where s is a reuced word. Clearly, if s is the empty word , this is OK. So, what I want to say is suppose s of length more than one. Then this will follow by the case when length is one. Suppose this formula is true for length one elements then for any element of length more than one, the part $(s_2, , s_k)$ remember, is never disturbed in the definition. So, you have to just put back those $(s_2, , s_k)$ part after getting the formula for the case when the length is equal to one. Alright? Therefore, you have to verify it only for length one elements.

(Refer Slide Time: 16:19)



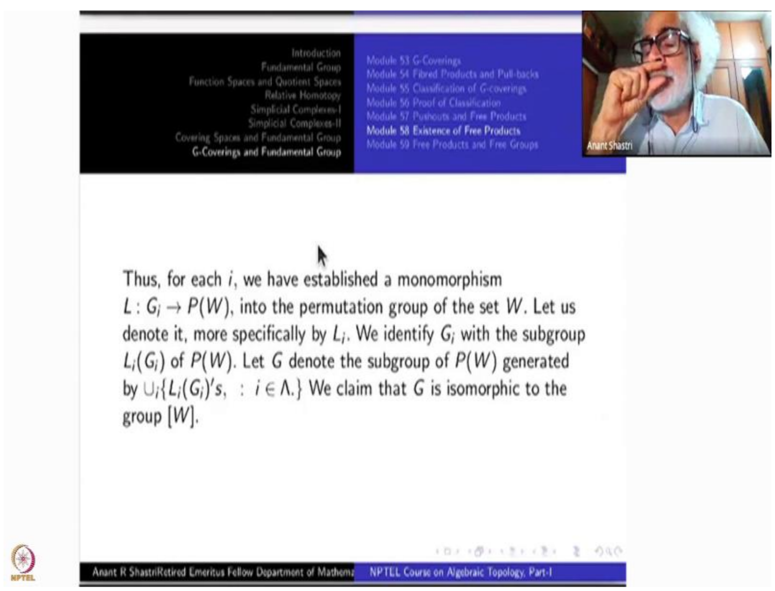
So, we have te check that $L_{gh}(s_1) = L_g(L_h(s_1)$. This is very easy.

(Refer Slide Time: 16:31)



Again you have to go through this list. For example, suppose first case namely s is empty that is actually a length 0 case. But L_h of the empty word is (h) of course $L_g(h) = (gh)$ because $g, h \in G_i$. Then after reduction if necessary it will be equal to L_{gh} operationg on the empty word.

(Refer Slide Time: 17:14)

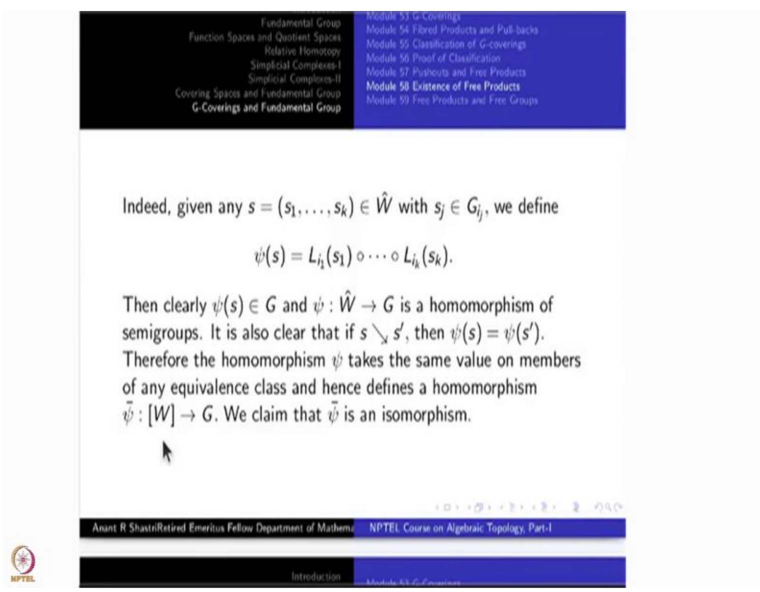


So, what we have done so far is that we have constructed for each $i, L_i : G_i \to P(W)$, that is a monomorphism. So, I can identify G_i with this subgroup $L_i(G_i) \subset P(W)$. So, this I can do for all i alright, I should need a different notation I will call them as L_i. So, this was, earlier, we had denoted by L, because I had fixed i. So let us call them L_i though such elaborate notation is not just necessary. What are all these subgroups in $P(W)$ give us? The union of all $L_i(G_i)$, they are each of them is a subgroup but union is not a subgroup. The union will generate a subgroup. That subgroup is isomorphic to this bracket $[W]$. That is what we want to show.

So, let us have a notation. This is final notation we are going to intrduce. Union of all the $L_i(G_i)$'s generates a subgroup of $P(W)$. and let us denote it gy G. So, our definition of G is this one now, what we are going to show is that this subgroup of $P(W)$ is actually isomorphic to this bracket $[W]$, the equivalence classes of free words, and we are also going to show that it is in biejective correspondence with the set of reduced words. So, various ways of looking at this group that will help to consolidate the entire thing.

So, this bracket $[W]$, we know is a group. On the othr hand, the set W of reduced words, we do not know that it is a group, but in each each equivalence class, there is a reduced word . We want to show that all these are isomorphic to each other. There is a group structure here. There is a group structure G here. You want to say these two are actually isomorphic.

(Refer Slide Time: 19:58)

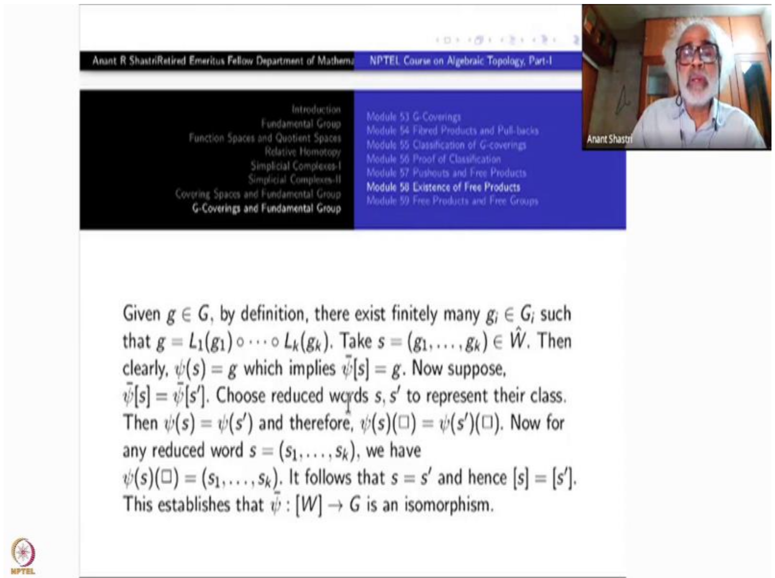


So, how do we proceed? Start with a free-word s inside $\hat{W}$. Now, you define a function $\psi : \hat{W} \to P(W)$. Indeed, we define a fnction which takes value inside G. $s = s_1 \ldots s_k$. Eeach s_i is in some G_{j_i} accordingly, take those monomorphisms L_{ji} and take $L_{ji}(s_i)$, compose them in the same order $L_{j_1} \circ L_{j_2} \circ \cdots \circ L_{j_k}$. They are all permutations of W, remember that, so you compose them. But they are all inside the union of all $L_j(G_j)$. Therefore, this element is inside G, the subgroup generated by this union. So, what we have got is for each element of $\hat{W}$ an element of G so, ψ is a function, this function is clearly a homomorphism of the semi group$\hat{W} \to G$ into a group. How do you compose on either side? $(s_1, \ldots, s_k) \circ (s'_1, \ldots, s'_r) = (s_1, \ldots, s_k, s'_1 \ldots, s'_r)$. On the other side also you are doing the same way. So this is a homomorphism of semigroups, of course the codomain is a group, but it is also a semigroup. One more thing you can say is that suppose s collapses to s', just a simple elementary collapsing, $\psi(s) = \psi(s')$. Why? There are two cases.

One of the s_is here maybe identity. Then $L_{j_i} L_{j\,i}(s_i) =:_J\, _i(1) = Id_W$ and the rest of them are the same on both sides. So the composition is the same.

Next suppose two of them say s_1, s_2 are consecutive letters which are in the same group G_j, then I am combining them into one letter $s_1 s_2$ but then $L_j(s_1) \circ L_j(s_2) = L_j(s_1 s_2)$ and the rest of the terms are the same on either side. Therefore, under elementary collapsing the value of $\psi(s)$ and $psi(s')$ are the same. What does this mean? This means that this map ψ actually factors down to give you a map $\bar{\psi} : [W] \to G$ from the quotient geoup of equivalnce classes to the group G. This is going to be an isomorphism.

(Refer Slide Time: 23:25)

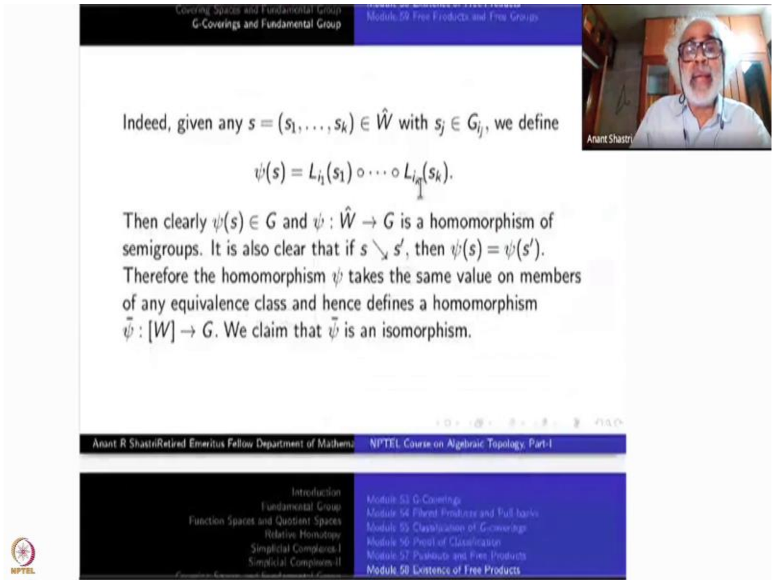


Let us see how. Every element $g \in G$ is a finite product of elements belonging to $L_j(G_j)$. So $g = L_{j_1}(g_1) \circ \cdots \circ L_{j_k}(g_k)$. Here $g_i \in G_{j_i}$ So, take the free word $s = g_1 \cdots g_k \in \hat{W}$, Then $\psi(s) = g$. Therefore $\bar{\psi}[s] = g$. What we have shown you just now is that $\bar{\psi}$ is surjective, because ψ itself is surjective.

Now, we want to show that this is injective also. Suppsoe $\psi(s) = \psi(s')$. You must show that $[s] = [s']$. Not $s = s'$ as free words, but the eqivalence classes. So, this is where we have to go to the classes. s may not be equal to s' but their equivalence classes are the same.

So, all that you have to do is to pick up reduced words which represent them. We have seen that in each class there is a reduced word take the reduced word representing them. Then $\psi(s) = \psi(s')$ because ψ takes the same value on each equivalence class. Now on W, these two endomorphisms are the same means, in particular, their value on the empty word must be the same. But by definition, what is the psi s of this one empty word?

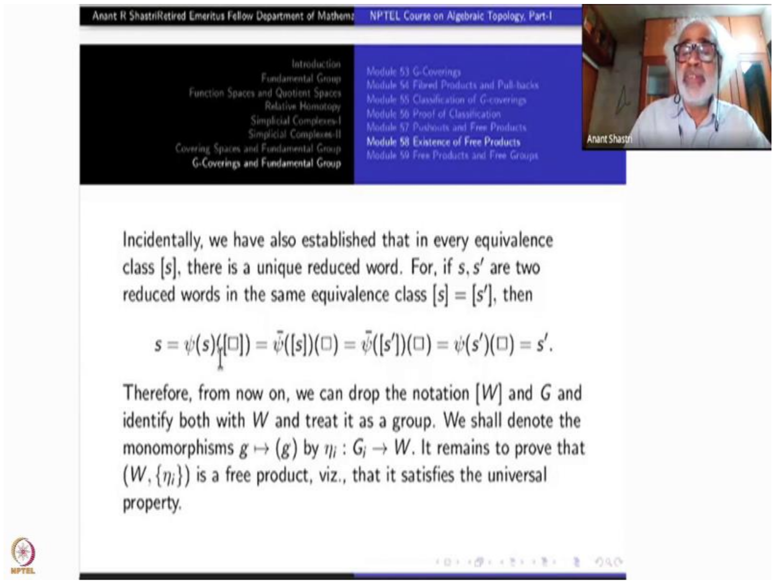
(Refer Slide Time: 26:20)



Suppose the reduced word is $s = s_1 \dots s_k$. Then $L_{j_k}(s_k)$ on the empty word will prodcue s_k. After that since s is reduced, succesively we get $L_{j_{i-1}}(s_i \dots s_k) = s_{i-1}s_i \dots s_k$. Finally we get $L_{j_1}(s_1)(s_2 \dots s_k) = s_1 \dots s_k$. Therefore the effect of $\psi(s)$ on the empty word is s itself. Since this true for the reduced word s' also, we conclude $s = s'$. Therefore $\bar{\psi}$ is injective.

So, you can directly do this by supposing $\bar{\psi}([s]) = Id_W$ and then showing that s is collapsible to an empty word. But that may be easier than what we have done here. There are different ways of putting this one idea. What we have done so far, is that $\bar{\psi} : [W] \to G$ is an isomorphism from the equivalence classes of words to this group, which is a sub group of the permutations of reduced words.

(Refer Slide Time: 27:45)



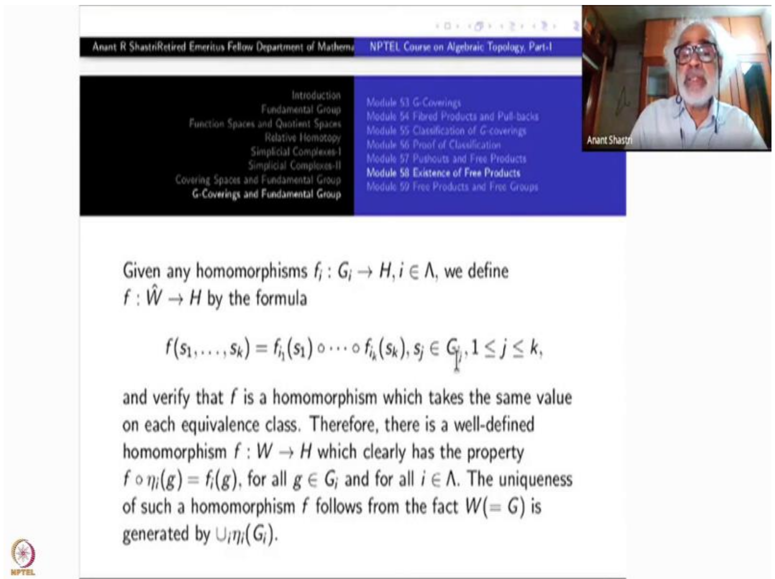
Already we have a hidden thing here, namely, in this group $[W]$. I wrote down this one not for fun. This already tells you that in each equivalence class, there is only one reduced word. Can you see that? This tells you that two reduced words if they are in same equivalence class, then they must be equal. So let us run through that proof again because this is so important.

Suppose on the contrary, that we have two reduced words in the same equivalence class. That is I am assuming $[s] = [s']$. We have see that for any redced word s, $\psi(s)$ operating on the empty word is equal to s. This much we have seen. We have also seen that ψ takes the same value on the equivalence class and used that to define $\bar{\psi}([s])$. Therefore $\psi(s) = \psi(s')$. This means then $s = s'$.

Therefore, in each equivalence class, there is exactly one reduced word. From that, it follows that $\bar{\psi} : W \to G$ is a bijection. You can now pullback the group operation of G on to W via $\bar{\psi}$. Defining a group operation directly on W would be a bit awkward. We hae avoided that.

Finally, if you want to just work with reduced words, what you have to do? Take a reduced word take another reduced word put one after the other. It may not be reduced but keep collapsing. You will go to a unique reduced word that is the definition of the composition law on reduced words you see why we needed to have uniqueness well before hand, otherwise definition of you know alpha composite beta, does not make sense. Even after this, proving associativity is hell of a problem. We avoid all this cleverly by going to the permutation group of all reduced words. Also we have shown that there are monomorphism from each G_i to this W now, is W is reduced word which is in bijection with G so I can replace this W by G if you want. So, we get these $\eta_i : G_i \to G$ is along with this G. This is going to be our free products so we are going to verify the universal property.

(Refer Slide Time: 31:16)

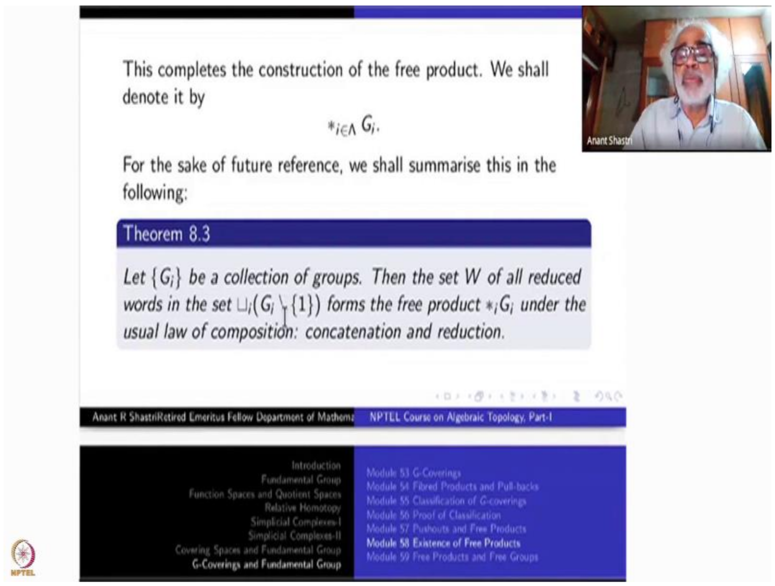


Given any homomorphism $f_i : G_i \to H$, for each i, a collection of homomorphisms, we define $f : \hat{W} \to H$, namely on the free words, by the formula $F(s_1, \ldots, s_k)$=$f_{j_1}(s_1) \circ \ldots \circ f_{j_k}(s_k)$, where $s_i \in G_{j_i}$. Take the corresponding f_{j_i} and their composition in the same order. There is no ambiguity in the free words, defining anything there is no ambiguity. verify that this is a homomorphism of the semigroup $\hat{W}$ to H. Exactly similar to the case of ψ.

The next thing is just like we have done for ψ, verify that F takes same value on each equivalence class, namely under elementary collapsing, the value of F does not change. Therefore we get a well defined map $f : W \to H$, which clearly has the property that if you take $f(\eta_i(g))$ for any $g \in G$, how we will define eta i of g, just singleton g that is a reduced word. So, this fi operating upon precisely that this sequencing is a singleton one that wherever it is that coming from Gi it is fi of that. So, $f(\eta_i(g))$=$f_i(g)$.

So, satisfies these conditions. Only thing is why f is unique? f is already determied on all the $\eta_i(G_i)$ which generate the group G. If you have a homomorphism of a group, it is completely determined once you know its values on a set of generators. That is enough to conclude the uniqueness. So, this completes the proof that this (G_i, η_i, G) is the free product of the family $\{G_i\}$.

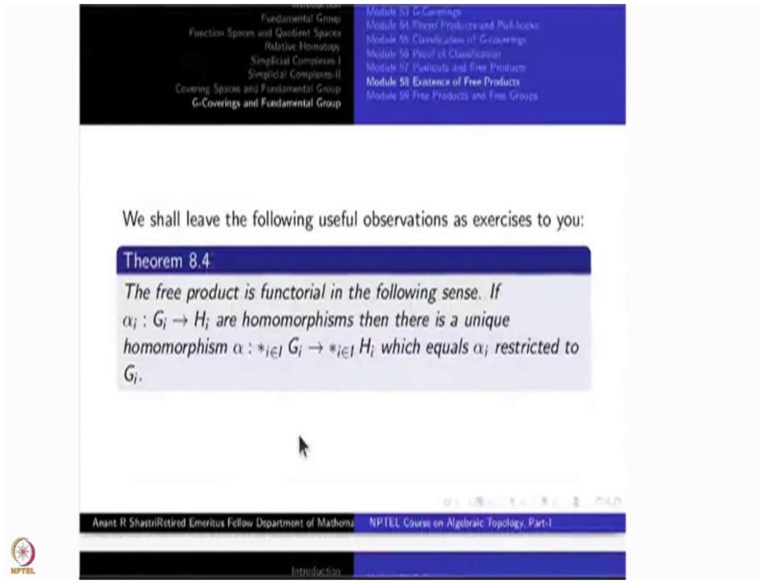
(Refer Slide Time: 34:51)



So, we have this notation for free product $*_{i\in\Lambda} G_i$ where Gi's is a families of groups with the indexing set Λ. So, we summarize this one whatever you have done so far. Start with a collection of groups. Then the set W of all reduced words in the union of these Gi's. Now, for reduced words you can take away the identity elements of each G_i, identity elements do not play any role, you can cut it way right in the beginning and take free word on $\bigcup_{\imath\in\Lambda}(G_i \setminus \{1\})$

So, just reduce words in these letters. Of course the binary operation is simply the concatanation of tow free words. So, we have not directly prove this we have proved it indirectly. Yeah, we have verified that this definition is the same as the other definition, but we have not verified that this definition is associative left multiple, left identity right identity those things we have (())(36:12).

(Refer Slide Time: 36:16)



Now, here is an easy result, if you have understood this construction, which is very useful namely, suppose you have a collection of homomorphism $\alpha_i : G_i \to H_i$. For the free product here then all these α_i's will extend to a single $\alpha : *G_i \to *H_i$.

If you followed this construction, then this is obvious, all that you have to do is look at the free words and define the function there in obvious manner. Then all that you have to do is keep reduction process. So, let us do all other constructions. like we have to push outs and various things. We have just done one single namely, construction of free products that we will do next time. Thank you.