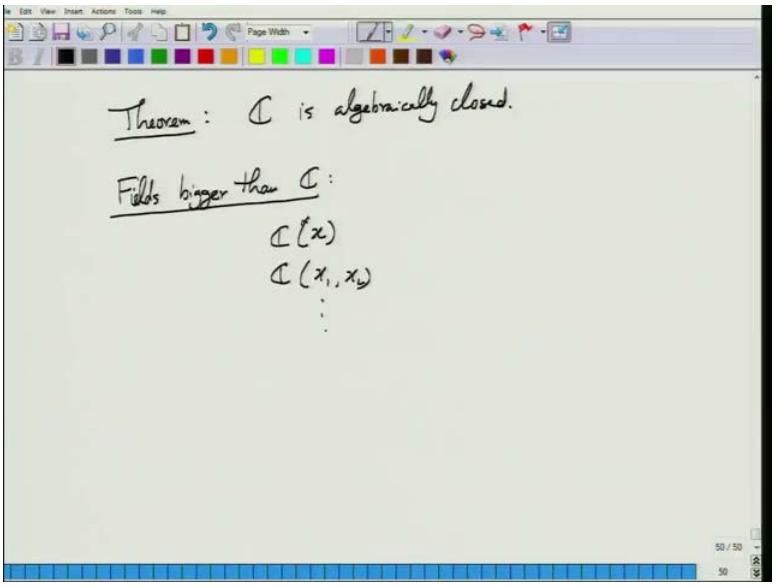
Modern Algebra
Prof. Manindra Agrawal
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

Lecture – 17
Finite Fields

(Refer Slide Time: 00:30)



So, in last lecture we defined algebraic closure of fields and this theorem I said, I am not going to prove. It is just a statement that said of complex numbers, which is a field, is algebraically closed, which means, that all polynomials defined over complex numbers have the roots within that, but this does not mean, that C is the biggest field that there is. There are fields which are bigger than complex number and the examples are easy to construct.

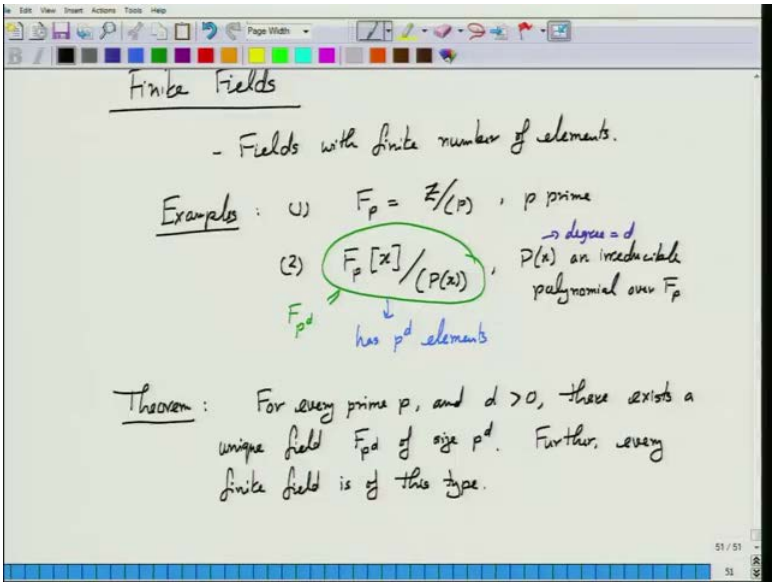
The fact, that C is algebraically closed only rules out one way of constructing, you know, extending a field. We can simply look at this field. This is the field of all rational functions in variable x with coefficients coming from complex numbers. This is the field and is certainly is bigger than complex numbers because there is no corresponding complex number to the variable x.

How you can formally prove it by showing, that there is no isomorphism between C x and C and then you can continue C x 1, x 2, this is even bigger and so on. So, there is really no such end or there is no such largest field because any field that you have, we

can always attach a fresh variable to it, look at the rational field with respect to that and then that is a bigger field. So, this is a chain which keeps on going, but we are again, this, these are two (Refer Time: 02:28) fields so to speak for us to consider.

We, in this course, we will now step back a bit and let me define a very interesting class of fields, which is called finite fields.

(Refer Slide Time: 02:43)



So, as the name suggests, this is fields with finite number of elements. This is not a used well field that you encounter, but we know, that their exist fields which are finite number of elements, F P, where P is a prime number is one such field, right. So, examples would be F P, which is simply Z quotiented with the prime ideal generated by prime number; that is certainly one. Can you think of any other finite field?

Student: (Refer Time: 04:00).

F P upon.

Student: (Refer Time: 04:05).

Yes, so, that is, I think your, the idea is correct. Look at the ring of polynomials in variable x over F P. Take a maximal ideal here. So, this is ring of polynomials in variable x. This ring I think we have already seen, every ideal is a principle ideal, right. So, if every ideal is principle ideal, then any prime ideal is maximal, that is very easy to see.

So, take any prime ideal here. A prime ideal will, it is a principle ideal, essentially it means, take an irreducible polynomial over F P and take the principle ideal generated by that polynomial. The P(x) is a polynomial with coefficients coming from F P and it is irreducible, that it does not factor in this ring F P [x], then the principle ideal generated by polynomial P is a prime ideal and maximal ideal. So, if we quotient this ring with the maximal ideal, you get a (Refer Time: 05:44).

What are the elements of the fields? We have known that as well. The elements are, if capital P is a polynomial, let us say, of degree d, then elements of this field will be polynomials of degree less than d, all such elements. So, how many such elements are there? P to the power d; every coefficient can take P values and there are d possible places. So, this is P to the d. So, this has therefore, P to the d elements and we write this field as F P to the d, just to highlight the fact that it has P to the d elements.

And then, there is a big theorem, which says, that for every prime P and number d greater than 0, there exists a unique field, which call F P to the d of size P to the d. So, this fields that are just listed out, these are unique, that is, there is only one or exactly one field of size P to the d. This is unlike again the infinite case, I mean, infinite case of course, when you look at there are many fields with infinite size, rational, reals and they are not isomorphic, whereas this saying all fields of size P to the d are isomorphic to each other. So, there is essentially only one. Further, every, every finite field is of this type; there is, there is no other finite field. So, this completely describes all finite fields.

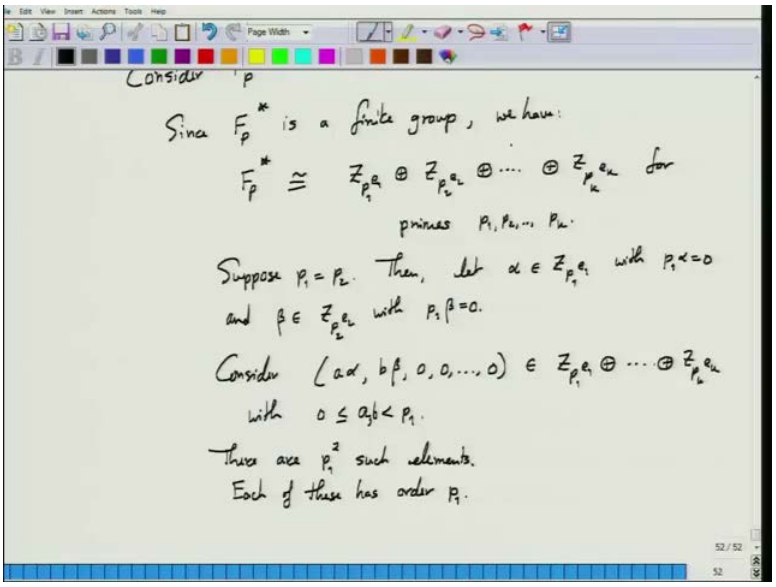
It is not difficult to prove except the uniqueness part. See, you can easily prove that every finite field must be of this type. Firstly, look at the characteristics of a field. We have already seen characteristic must be a prime number in the sense, it is finite, the characteristic cannot be 0 and if we look at 1 plus 1 plus 1 plus 1, eventually after finite number of additions you will repeat and therefore, that sum will become 0. So, its characteristic has to be a prime number.

Secondly, since the field is finite, if you said the characteristic is P, then you start with F P, which lies inside the finite field and then, every other element of the field is algebraic over F P, again because it is finite. If you look at every other element, say, alpha and look at alpha, alpha square, alpha cube, alpha to the 4, alpha to the 5, successive power after finite even powers, you will have again a repetition, that is, power will be a linear

combination of previous powers and so, every other element is algebraic. And then, therefore, this field is an algebraic extension of F P and then one can show, that there is an irreducible polynomial in F P x so that if you quotient that irreducible polynomial with this, you will get the field.

So, that is the broad outline of the proof, but I do not want to give the full proof. It is really not necessary, important thing is this theorem, allows us to completely characterise finite fields.

(Refer Slide Time: 10:51)



Now, the finite fields behave in a way, which is at times very different from normal fields. One prime example of this is this group F P star. This, if you recall, is the set of all non-zero elements of F P and we, look, this set under multiplication, it forms a group under multiplication, it is a finite group, what is the structure of this? Now, just bring in your knowledge about finite groups into play here. We had this, right, when we were discussing group theory I gave this theorem which describes a structure of finite groups. Do you remember that theorem? A theorem said that the any finite group is isomorphic to.

Student: Groups of prime.

Yes. So, prime and prime powers Z sum of direct sums of Z P I to the e I for various primes and numbers, right. So, let us start from that fact. Since F P star is a finite group,

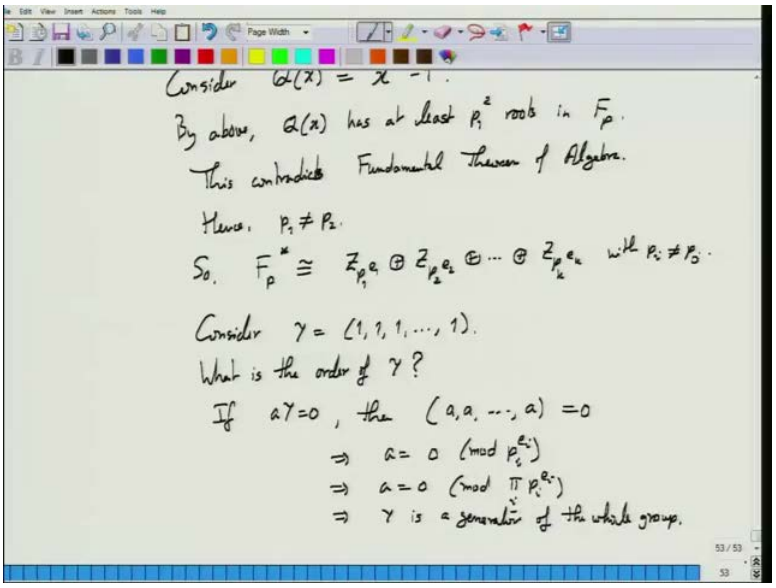
we have F P star is isomorphic to Z P 1 to the e 1 P k to the e k. I am writing this additively, whereas this is the multiplicative group, but I hope you understand the meaning. This direct sum is to be treated as a, just a direct sum, so just a formal operation. The group operation is treated as the multiplication of (Refer Time: 13:21).

Now, the next question, that I am going to ask is, is, are these primes distinct, P 1 to P k? In the finite group structure they need not be distinct, but in this case are they distinct? And the answer to that is yes, they are distinct. Why? Well, let us, here is a very simple trick.

Suppose P 1 equals P 2, then let alpha, I am going to pick two elements; alpha and beta are the following kind. If you look at this group, Z P 1 to be e 1, this is a cyclic group with one generator. The generator being number one and the size of this is P 1 to the e 1. So, then there exist an element in this whose order is P 1, namely their element would be in this group, would be P 1 to the e 1 minus 1, that is one such element. So, instead of calling it P 1 to the e 1 minus 1, I am just calling it alpha and if there is an element alpha whose order is P 1 in this. Similarly, in Z P 2 to the e 2, there is an element P 1 equals P 2 because there is an element beta with that order as P 1.

Now, consider, consider these elements a alpha comma b beta and all 0s, that belong to this direct sum where a, b both range between 0 and P 1. How many such elements are there; a and b both takes exactly P 1 values from 0 to P 1 minus 1. So, there are exactly P 1 square such elements and each element has order P 1 and this is isomorphic to F P star.

(Refer Slide Time: 17:19)



So, now, let us switch attention to F P star and let me ask the following question. Consider this equation Q(x) x to the P 1 minus 1, how many roots does this polynomial have in F P star? Since this is isomorphic to this, we can ask the analogous question here that how many elements are here whose order is P 1. So, this x to the P 1 minus 1 is 0 means, x to the P 1 is 1, which means, whatever x that satisfy as an order P 1 in the multiplicative group.

So, analogously we ask, since this is isomorphic to this, how many elements of this have order P 1 more, at least P 1 square and say there is an isomorphism? So, each one of those P 1 square elements will give rise to a unique elements here in F P star. This means Q has at least P 1 square roots in F P. Is this possible? The fundamental theorem of arithmetic, this is a field, this is a polynomial degree P 1, can have at most P 1 groups, it cannot have P 1 square roots. Fundamental theorem of algebra, hence P 1 is not equal to P 2. So, F P star is now, we can say, isomorphic to Z P 1 to the e 1 P 2 to the e 2, P I not equal to P j.

Now, consider the following element, gamma equals 1, 1, all 1s, what is the order of gamma in this group?

Student: 1.

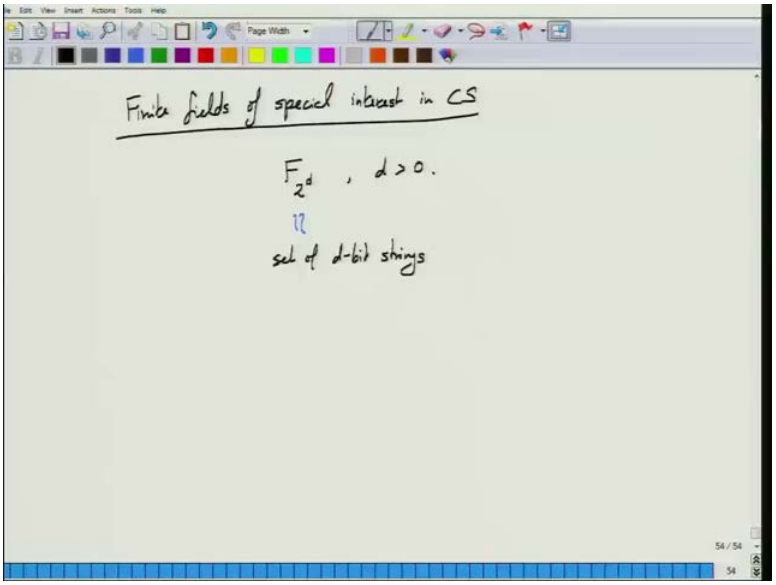
Order means, the identity of this group is all 0s. So, order means, that how many times does this element need to be added to itself to get a 0?

Student: multiplication of whole primes.

If, let us say, if a gamma is 0, then we have a, a, a is 0. This implies, that a is 0 modulo P I to the e I; that is the only way because that is a very simple way. And since each P I is a distinct prime, this implies that a is 0 modulo product of I p to the e I. And what is this product size? This is exactly the size of the whole group. So, this implies, that gamma is a generator of the whole group and since gamma is a generator of this group, there is a corresponding isomorphically generator of F P star, which means, that F P star is cyclic.

So, there is only one generator that will generate the entire F p star, which viewed in terms of F P star means, that there is an element, let us say, g in F P star such that when you consider successive powers of g, g, g square, g cube, that list outs all non-zero elements of the field. This is in contrast with, let us say, Q star. The q star does not have finite number of generators itself. Since there are infinitely only primes, each prime and its power, essentially, is one generator; each prime is one generator. So, its successive powers and, but there are infinitely many generators, whereas for a finite field, exactly one generator in the mulplicator. So, finite fields therefore, have a much simpler structure compared to the infinite fields.

(Refer Slide Time: 22:57)



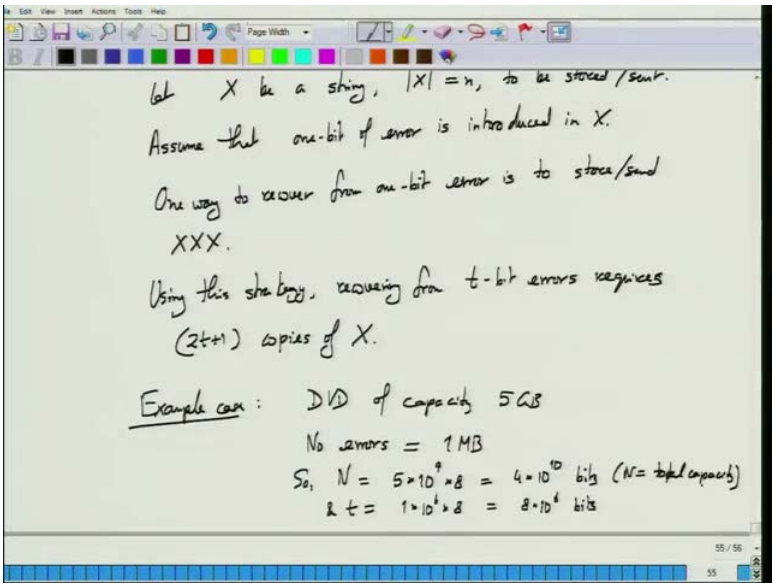
And one particular type of finite fields has been used very widely in computer science, which is this. Take P equals to F 2 to the d. So, (Refer Time: 23:19) exactly 2 to the d elements. And the reason they are all special interest is, because in computer science, whenever you write any data, you write it as a bit string. So, since this field has 2 to the d elements exactly and if you think each elements is essentially a polynomial of degree less than 0 with coefficients being 0 or 1, that is exactly analogous to a d-bit strings. So, this is analogous to set of d-bit strings and often it is very useful to associate this meaning with d-bit strings that they are elements of F 2 to the d because then, you can do arithmetic over it.

So, let me give you one example of this, which is an application without which modern life would not be possible. So, you do this, now what is one of the cornerstones of the modern life? Communication, right, all kinds of communication, which happens over radio waves and microwaves and all kind of waves; not only that, there is a and a, we also, another very important facet is storage of data. You know, we store enormous amount of data.

Now, one problem common to both of these applications is, like if you are storing a data, what if some part of the data gets corrupted? Similarly, if you are sending data from one place to another over, let us say, radio channel, what if along some part of that gets corrupted? Then, if the receiving end you have that corrupted data, which is not that useful at times. Similarly, if you are reading the data which is stored and it is corrupted, then may not be that useful. In fact, that happens, similar storage, in both the application it happens repeatedly. In communication channel, radio waves, there are always going to be disturbances, no, electric disturbances or any, all kinds of disturbance, which will bring in some amount of distortion in the data.

In the storage, I mean, if you look at the CDs, DVDs, where the data is stored, and I am sure each one of you yourself has put scratches on those, the back surface where the data is stored; putting a scratch means the data stored there is erased. And so, it is important, that in such a situation we can still recover exactly what was sent or what was stored. So, how is that to be done? Do you have any idea that comes under the broad field of error correction; the question is how do we do error correction?

(Refer Slide Time: 27:04)



So, we can model it, as the entire data is a long string, let us call it X of size n and that is to be either stored or sent. Now, at the receiving side we may have errors or after, after reading we may have errors. So, how do we get rid of this? And let us start with the simplest case, that there is one bit of error. Now, my target is to look out from the error. So, how should we store this X or how should we send that X so that finally, that error is recovered, one bit? Say some solution.

Student: (Refer Time: 28:45)

Let us, I give you a very simple solution, send X again. There is two copies of X. So, instead of sending X or storing X, we store X X. So, then, one bit of error means, one copy of X gets corrupted. So, suppose, suppose 5th bit of X gets corrupted, but the other X will have the correct 5th bit. So, can we recover from this, the correct 5th bit value? We would not be able to because then, what when we read, we will see the 5th bit in two copies, in one of them it is 0 and one of them it is 1. The question is, which is the right value? We do not know which actually bit got corrupted. So, send, storing two copies of X is not good enough, sending two copies.

But let us say, send 3 copies of X and then one bit gets corrupted. Now, we are fine because then, we have 3 copies of X, look at the three copies of 5th bit, two of them will be uncorrupted, one of them will be corrupted. So, two values will be original values, the 3rd value will be the wrong one. So, pick given, so we will either have 0, 1, 1 or 1, 1, 0,

so pick the majority occurring bit value that will correct from 1-bit error; that is very simple scheme. So, that is one way of recovering from one bit of error.

See, the important thing is here is, that we, after receiving we do not really know where the error is occurred, the only thing we know is that some error may have occurred. And we are, let us say a priory told, that look, at most 1-bit of error can occur, then we can recover. But this is really a toy example, I mean, we should expect much more than one bits of error to occur.

Student: (Refer Time: 31:22).

Yes. So, that means more than one bit of error. So, if let us say, two bits of error has occurred, then X X X would not work, because.

Student: We know one bit of error.

OK.

Student: Getting in two messages.

Yes.

Student: Both the messages come at the same place.

OK.

Student: So, (Refer Time: 31:52) taking the majority.

Yes.

Student: (Refer Time: 31:53).

Why? So, our X X X is my message and we are assuming that one bit of error will occur somewhere here.

Student: Same bit error (Refer Time: 32:02).

How can it occur? It was one bit of error can get corrupted. In X X X, almost one bit will go wrong. I am not saying that in each copy of X one bit can go wrong, but this is, like I said, a very toy example. In real life, the errors will be many more. So, if you say, two

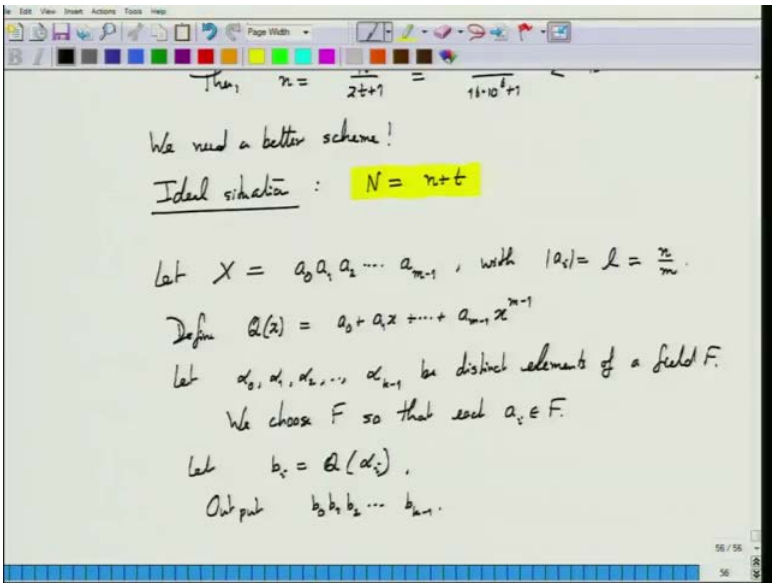
bits of error, which is still a toy example, even with two bits this X X X strategy would not work. Just like you pointed out, if two bits in the same location of two copies of X that gets corrupted, then we would not. So, to recover from two bits of error, we have to actually use 5 copies of X.

So, in general, using this strategy, recovering from t bit errors requires 2 t plus 1 copies of X because in the worst case, those t errors can be in the exactly the same position of t copies of X and then we need t plus 1 more copies, which is the majority copies where that error has not occurred and that ca be written. So, this is a possible scheme. The only thing is, that this is not really a practical scheme.

If you consider, that we have to, let us take for our calculation the example of a DVD. What is the capacity of a DVD? About 5 GB, right, typically and suppose we want to recover from and one scratch. Although it may be thin, because the data is so densely packed, one scratch can really erase large number of bits, tens of thousands of bits and there can be multiple scratches also.

So, shall we say that we have, let us say in the example case, DVD of capacity 5 GB, number of errors, let us given in number to this, let us say 1 megabyte. So, multiple scratches are allowed in here, 1 megabyte is a reasonable number, still compared to 5 GB is a very tiny number. So, n for us is 5 into, gigabytes is 10 to the 9 into 8, it is a byte, so number of bits would be 4 into 10 to the 10 bits, and t is 8 into 10 to the 6 bits. So, these many bits can go wrong and this is the total storage capacity available. Then, if we implement this scheme which I just described, this is the final capacity. So, what would be small n?

(Refer Slide Time: 36:31)



If you have total capacity is given and number of errors bits is given and we are implementing this earlier strategy, small n is equal to capital N divided by 2 t plus 1, correct. Capital N is 10 in, 4 into 10 to the power 10 divide by, this is t is 8, so 16 into 10 to the power 6 plus 1 and this is less than, certainly less than 10 to the 4, that is, 10000 bits. So, if you implement this claim, you can, in the whole DVD of 5 megabyte size, you can store data worth 10000 bits, which is a little more than 1 kilobyte. So, this is a completely worthless scheme if you want to practically implement this. And the same situation will occur in transmission; I mean the same thing. I mean, you transmit enormous amount of data to, in order to send some small amount of real information; that is unacceptable. So, we need a better scheme.

What is the ideal situation? So, we have n, small n, bits of data, t errors we want to tolerate, what would be the ideal situation? That, see, if t bits get erased and you still want original n bits to be present. So, you need at least n plus t bits. So, ideal situation is, that capital N is small n plus t, cannot hope to achieve better than this at all that can even come close to this. That is the question and the answer is yes and only, it is only thanks to finite fields, that we can come, very close to this.

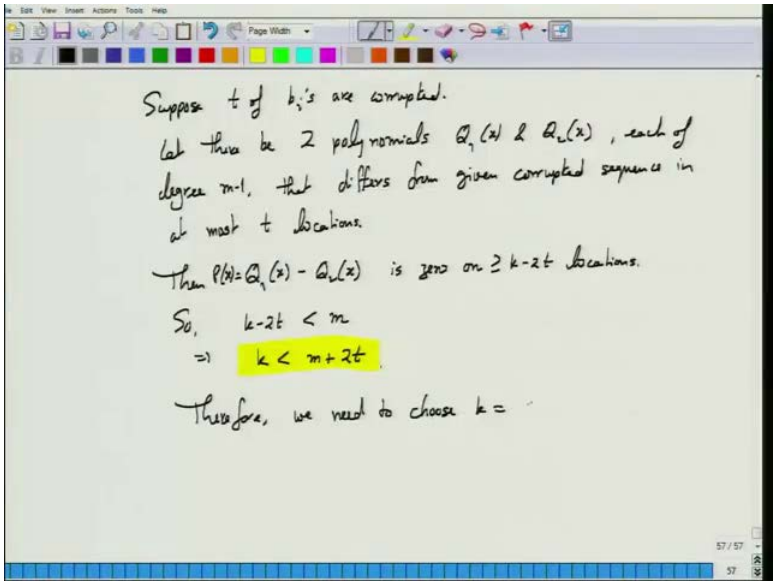
In fact, we can, capital N we can set, we can come to n plus 2t, just a little bit more. So, how do we achieve that? So, for that let us use some more knowledge that we have developed. We have been talking about polynomial square bit and the fact about the

roots; I mean the fields, and so on. So, let us do the basic set up in this lecture and then we will continue tomorrow.

First thing first, that let us first chop x into small pieces, a 0, a 1 to a m mins 1 with each length of each a I being equal to l, which is saying, it is going to be m pieces, it is n by m. Now, define a polynomial Q(x) as this. The coefficient of the polynomials are coming from the message, ok. Now, we let alpha 1 or alpha 0, alpha 1, alpha 2 to alpha k minus 1. I do not think I have used k anywhere, be distinct elements. So, let alpha 0 to alpha k minus 1 be distinct elements of a field F. And this field F will choose, we will also, we chose F so that each of a I is also in F.

Now, we let b I equals Q of alpha a and then, output b 0, b 1, b 2 up to b k minus 1. So, the original string or message was a 0 to a m minus 1. I have done this transformation of that message and eventually got this b 0, b m minus 1 k minus 1 and this I am claiming is the coded message. So, this is what is transmitted or stored.

(Refer Slide Time: 42:38)



Now, b of them will get corrupted. So, let us suppose t of b I's are corrupted and still I want to recover the original message, which is a 0 to a m minus 1. (Refer Slide Time: 36:31) That was associated with this polynomial, so that is equivalent to the polynomial Q because coefficient of polynomial Q gives exactly the message. So, now let the question, that I need to ask is, I have, I am given this sequence with t up, up to t of them corrupted, is there a polynomial Q?

So, I do not know what the polynomial Q is, I only know this. Is there a polynomial Q such that this evaluations of Q on alpha 0 to alpha k minus 1 produces values such that at most t of them differ from the given values, that is, if I can complete the polynomial, then I have recovered the messages unless there are more than one polynomials. If there are two polynomials or three polynomials of this property, then I have a problem because then, I would not know what that original message is. So, let me ask the following.

Then, this polynomial, see Q 1, Q 2 differs from the given sequence in at most t locations. Then, then if you look at the Q 1 sequence and Q 2 sequence, there they will differ on at most two t locations. So, therefore, Q 1 minus Q 2 will be 0 on at least k minus 2 t location. When I say location, I actually mean these values, b 0 to b 1, on at least so many locations. So, this is the polynomial, P is the polynomial of degree m minus 1 and it is 0 on at least k minus 2 t values, which means, that k minus 2 t is less than m or in other words, k is less than m plus 2 t.

So, which means, if I choose k to be m plus 2 t, then there cannot be two polynomials. A k has to be less than m plus 2 t in order for two polynomials, Q 1 and Q 2 to differ with the given sequence at most t locations. And if I choose k to be m plus 2 t, there cannot be two polynomials, there will be a unique polynomial and that unique polynomial will be the one, you know, the Q which is the original one. So, I can recover that. So, therefore, we need to, we need to fix; we need to choose k equals m plus 2 t, fine.

So, let me stop here.