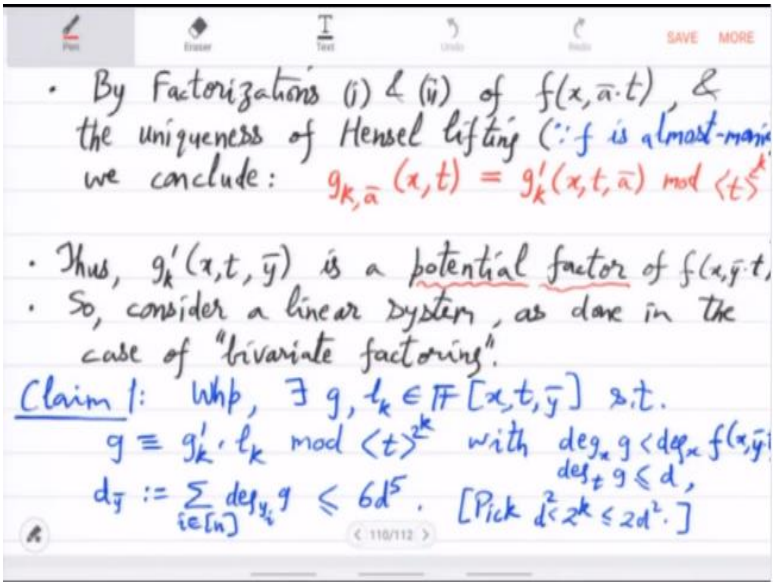
Computational Number Theory and Algebra
Prof. Nitin Saxena
Department of Computer Science and Engineering
Indian Institute of Technology-Kanpur

Lecture – 19

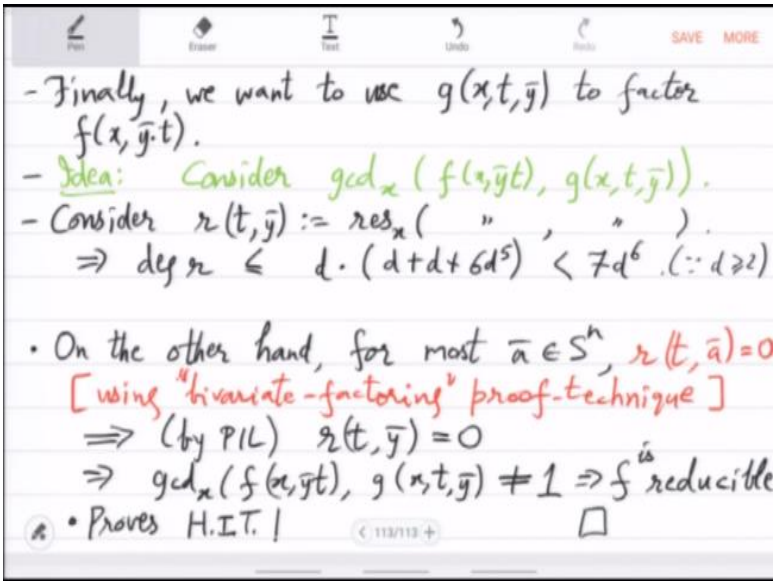
(Refer Slide Time: 00:15)



Okay, last time we were finishing the proof of Hilbert's irreducibility theorem and the place where we were is this claim 1 where we are trying to show that assuming that for an a bar you have factored f x, a bar t mod t power. Hensel lifting will give you this factors g k prime which is in formal y bar. And then from g k prime we want to now actually get to a factor of f, actual factor of f.

So first the claim was claim 1 is saying that there will be a polynomial g with limited degree which is a multiple of g k prime. Okay that we have shown. So the way we showed this is by saying that well there is a g for random fixings of y bar. Hence, there is also a g for formal y bar. So once we get this g now the next step will be taking its GCD with f.

(Refer Slide Time: 01:27)



So finally, we want to use g (x, t, y bar) to factor f (x, y bar t). Okay. So idea here is as I said take GCD. So consider GCD of f with g and the claim is that this GCD will actually give you a factor of f okay. So why is that? So to analyze that situation we consider the resultant. So resultant x will be eliminated. So you only have t and y bar. So this is resultant of f with g. And note that the degree of r is at most.

So this will be degree of f with respect to x is d. And we also know that degree of g with respect to x is less than d. So you are looking at the determinant of around 2d cross 2d matrix. So this is definitely smaller than 2d times, what are the what is the degree of the entries of or coefficients of f with respect to x monomials, right. So they are actually themselves polynomials in y bar and t.

So you have to use the degree bounds of g here. So let me write that. So recall that with respect to x it is d. With respect to t it is d. And with respect to y bar it was 6d 5. So which we can say is at most 7d 6, right. I think this 2 is unnecessary. This is actually d. You are looking at I think this d cross d matrix. So this comes out to be, the dominating term here is 6 d to the 6. And plus there are these lower order terms 2d square.

So that you can see is overall less than 7d 6 since d we are assuming to be at least 2. So this is a strict upper bound. Okay. Now on the other hand for most a bars you know that r at a bar vanishes. Why is that? So r vanishes at y bar equal to a bar for most a bar because f and g have a GCD. f and g have a non-trivial GCD because you

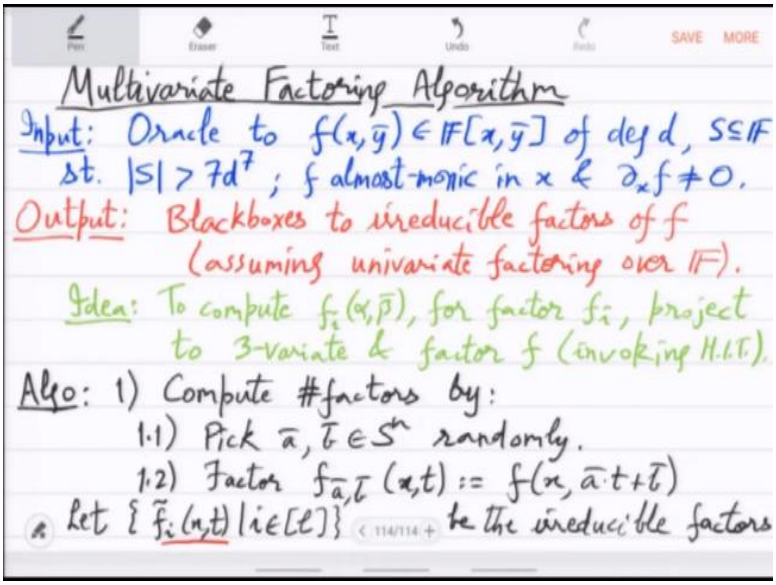
constructed g by this, you first did Hensel lifting then you solved a linear system of equation, right.

And recall the proof of bivariate factoring. So from that you will get, right. So you know that g in fact is of will give you a factor with f for y bar equal to a bar. So hence the resultant will be zero. So actually most of the a bars they are a root of r. But when that happens, see r has limited degree, right. If it has too many roots, then by the polynomial identity lemma you deduce that r is actually zero, that r t, y bar is 0, which means that there is a GCD.

So which means that f is reducible. Okay, so f is reducible and this proves Hilbert's irreducibility theorem, right. So what we have shown is that, if you start with f being reducible and square-free for most a bars, then f is indeed reducible for a formal for the formal y bar, okay. Contrapositive is that if you start with f irreducible then for most a bars it remains irreducible, okay. And you also get the specific probability bounds.

So this probability, error probability is 7d 6 divided by the sample size. So we have proved this theorem HIT. So this theorem is now proved. So using this theorem it is not very difficult to come up with the multivariate factoring algorithm.

(Refer Slide Time: 09:01)



Okay, so let us go through the steps in detail. So what is the input? So you are given an Oracle to this polynomial x, y bar over some field. You can assume degree is d.

Then you can assume that you have a sample space s of size large enough let us say 7d 7, is slightly more than 7d 6. We can assume f to be almost monic without loss of generality. And you can assume that the derivative does not vanish, okay.

Now in the output what you will get, you will get black boxes or Oracles to irreducible factors of f assuming that you can factor univariates over f. over the base field f. Okay, so if you do not know how to factor univariates then obviously, you will not, you will never know how to factor multivariates.

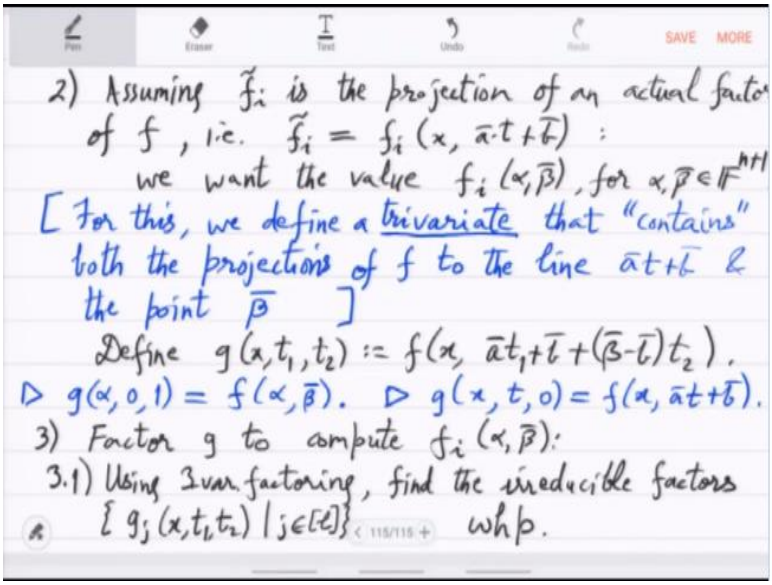
But this algorithm is the way to factor multivariates and at the black box level working with black boxes as long as you can factor univariates. Okay, so the idea is just project to trivariate and using Hilbert's irreducibility theorem get the let us say the value. So to compute f i alpha, beta bar for factor f i project to trivariate and factor, okay. So here you will be invoking irreducibility theorem.

So you are given a point alpha, beta bar and you want and you are given and this number say i which basically points to the factor f i in which you are interested and you want the value of f i at alpha beta bar. So to compute this, you do not know f i. So the way you will do this is you will first project f using alpha, beta bar to trivariate three variables. By Hilbert's irreducibility theorem there will be a one-to-one correspondence amongst the factors.

Three variate factorization, you will do by Hensel lifting method. You will get projected f i and from that you will compute, we will see that you can easily compute f i alpha, beta bar, okay. So let us implement this idea in great detail, okay. So first compute the number of factors. So pick a bar, b bar randomly and project to bivariate. So factor this bivariate, right? It is x, a bar t plus b bar.

So that is the bivariate projection as used in Hilbert's irreducibility theorem in its proof, factor this. So let f i tilde be the l many factors, okay. So you have obtained these factors. This can be done efficiently because you just have to factor this bivariate f a bar, b bar. That you know how to do as long as you know how to factor univariates.

(Refer Slide Time: 15:55)



Next what you do is assuming f i tilde is the projection of an actual factor of f that is f i tilde equals f i x, a bar t plus b bar. We want the value f i alpha, beta bar. So for a given alpha, beta bar you want the value f i alpha, beta bar. You do not know f i but you do know f i tilde its bivariate projection. So in this binary projection you can substitute x to be alpha, but it is not clear how will you, what to substitute in t so that you get beta bar, right? That may not be possible.

So to get around that problem, you have to do more computation. So we define a bivariate. We define a trivariate that contains both the projections of f to the line a bar t plus b bar. And it also contains the point. And the point beta bar, okay. So that is why trivariate is needed, because we want not only this line a bar t plus b bar to be contained in it in the projection, but also we want the point alpha, beta bar.

Because only when the point is present there can we evaluate f i at that. So let us give the definition. So g (x, t 1, t 2) is so you want the line, right? So a bar t 1 plus b bar should be there. And you also want the point. In this you want also the point beta bar. So that is plus beta bar minus b bar t 2, right. So note that this works. So g (alpha, 0, 1) right this is f at alpha, beta bar.

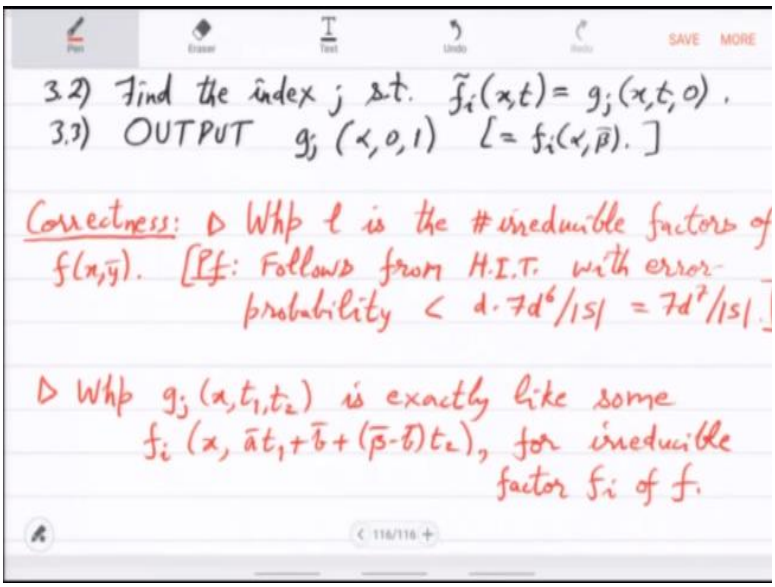
Moreover, so it contains the point beta bar and it also contains the line because that is clear because you get g (x, t, 0) right what is that? That is just what you had before x, a bar t plus b bar, right. So g is this trivariate projection in which you have the line

contained like before and you also have the point contained and by Hilbert's irreducibility theorem, there is a bijection if you look at the set of irreducible factors.

So whatever were the set of irreducible factors of f there are in one-to-one correspondence with those of bivariate projection which are in one-to-one correspondence with those of trivariate projection. So we just have to do it again. We just have to factor g, okay and we are done. So factor g to compute f i alpha, beta bar for any given alpha, beta bar. If you want details simple now. So basically you do three variate factoring.

Find the irreducible factors g j (x, t 1, t 2). You will get this with high probability. So g j are the factors of the trivariate. And then by looking at g j and f i tilde you will be able to set up the correspondence, okay. Which g j corresponds to which f i tilde? That correspondence you will set just by inspection.

(Refer Slide Time: 23:15)



So f i tilde x, t should be equal to g j (x, t, 0). Once you have identified it g j then you can output g j (alpha, 0, 1), okay. This is equal to as observed before, this is equal to f i alpha, beta bar, okay. That is the value you were looking for. So for correctness I am just making the following remarks. It is easy to see that the whole mainly because of Hilbert's irreducibility theorem.

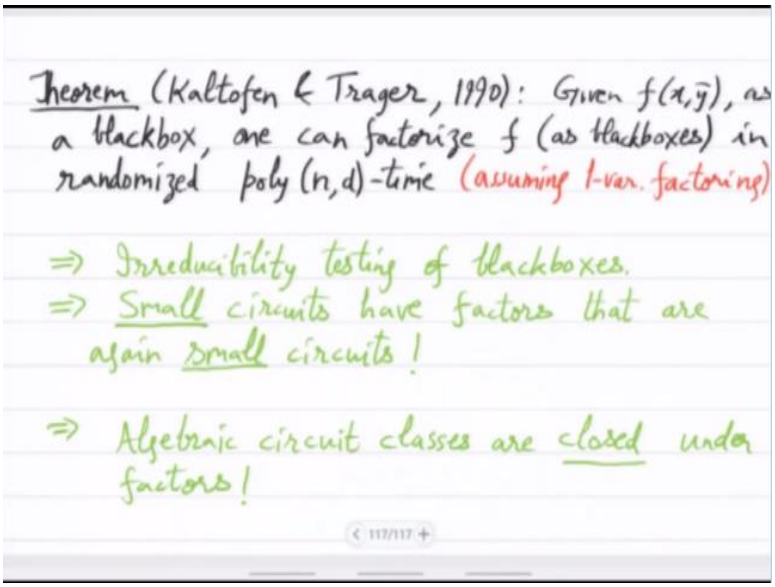
So first observation is that with high probability l is the number of factors, number of irreducible factors of f x, y bar. Why is that? Well, so this follows from Hilbert's

irreducibility theorem, right. Because pick any irreducible factor of f, let us call it f i, under the bivariate projection that we have used f i will also remain irreducible, right. So the irreducible factors of f will remain irreducible when you go to bivariate.

And so the information l is not lost. That you can see. And what is the probability? So with error probability so you get 7d 6 by s for one irreducible factor. And there may be d, maximum d irreducible factors. So you have to multiply it by d that is the error. Okay, that is why we need it slightly bigger, slightly larger s. So this 7d 7 by s is the error probability.

Second observation is which is important for the correctness is that this g j that you got g j is x, t 1, t 2. It is a trivariate irreducible factor. g j is exactly like some actual factor f i projected for factor f i of f, okay. So that is the algorithm. You project down to bivariate, get the number of factors, then you project down to this trivariate. Again factorize and set up the one-to-one correspondence with the previous set of irreducible factors and then you get the value of each of the factors at alpha, beta bar. Okay, so I hope it is clear.

(Refer Slide Time: 29:12)



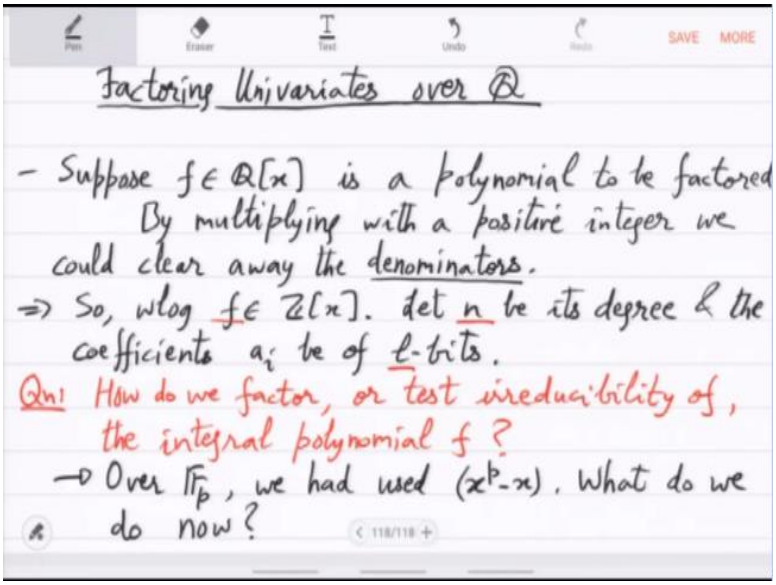
So what we have shown now, after all this hard work what we have shown is the following classical theorem by Kaltofen and Trager. It says that given f as a black box one can factorize f as black boxes in randomized polynomial time. Polynomial in what? Polynomial in the number of variables and the degree assuming that you have done you have solved univariate factoring, okay.

So this is a major result in computational algebra as well as computational complexity because this is saying that even Oracles can be factored into Oracles, right and remember that a polynomial given as an Oracle can have or usually has exponentially many monomials. So and the factors also could have exponentially many monomials but there is still a fast way to do this and give present the factors efficiently.

It goes without saying that this is also the first irreducibility test, right. You can test black boxes for irreducibility. Also, this can also be the proof can also be seen or slightly modified to show that if you are presented f as a circuit then its factors are also small circuits, okay. So small circuits, circuit classes are closed under factorization, under factors. So it is basically a closure result for algebraic circuit complexity, okay.

So at this point we will move, we are actually done with factorization, the general methods for factorization. We will now change the field. Now look at this kind of most natural field, field of rationals and over that look at univariate integral polynomials. So can we factor them? Can we test their irreducibility? Okay, we will now study that question and that is a kind of a different area. It leads to very interesting techniques.

(Refer Slide Time: 34:29)



Okay, so what you are given is a integral polynomial or let us say a polynomial with the rational coefficients, a rational polynomial. Suppose f is a polynomial to be

factored. Now note that at this point it is not even clear whether we can check whether this polynomial is factorizable, right. Even checking question is not clear. But what is clear is that we can simplify it a little. We can make the coefficients integers.

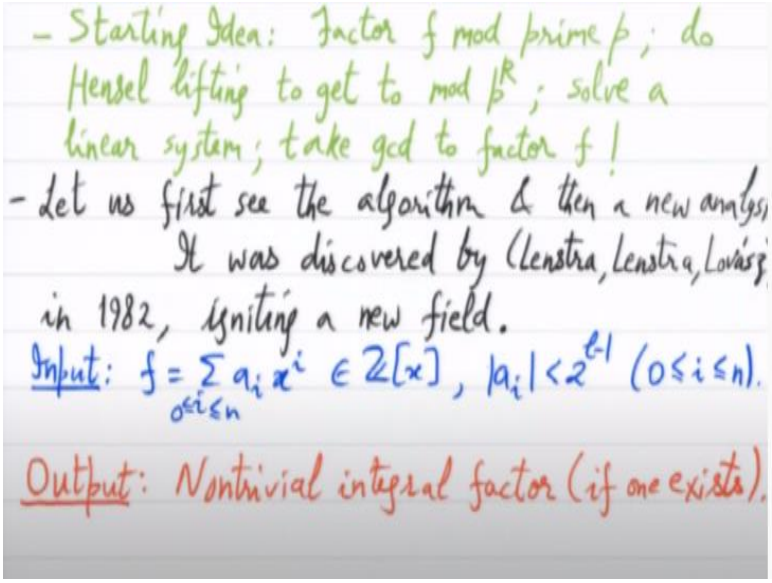
We could clear away the denominators, right. So without loss of generality can assume f to be integral, right. If there was a fraction like 1 by 3 appearing then you can multiply by 3 and it will become 1, right. So all the coefficients can be made integers by taking let us say LCM of the denominators. And then you can check that integral polynomial is irreducible if and only if the original polynomial was irreducible, right.

That you can check. It is not difficult to show. Let n be its degree and the coefficients a i be of l-bits, okay. We will work with these two parameters n and l. So to present f you need around nl n times l bits. That is your input size. So our question is how do we factor or test irreducibility of the integral polynomial f, right. So this is not a very easy question.

The only factoring technique that you have seen till now was in the case of finite fields, but there you used the Frobenius, right the Frobenius map. So this x to the p minus x was used. Now that has no applicability here. So over F p we had used x to the p minus x. But what do we do now, right? That is not clear. There is no Frobenius equivalent here. So instead of trying to solve it directly we will instead try to relate it to mod p factorization, okay.

So look at f modulo prime and ask the question whether f mod p is reducible. So note that if f mod p is irreducible then f is also irreducible, right. So there is some connection, but it is not complete because it is possible that f was irreducible, but f mod p it is reducible, it factors. So there is not if and only if, but there is something that we can try to build on. So that is the starting idea.

(Refer Slide Time: 41:06)



So factor f mod prime p. You find the factors, then you can try to do Hensel lifting. If you get coprime factors then you can do Hensel lifting. And let us see you get to p raised to k. So now you have factors of a f with in large modulus, p raised to k. But still these factors may not be actual factors of f, right. So at this point you try to do what you did in the bivariate factoring idea that you set up a linear system, try to solve it.

Solve a linear system and then take GCD, right. These are the steps. This is what was happening the main steps that you had when you went from bivariate factoring to univariate factoring. In fact, this is also what we did to prove Gilbert's irreducibility theorem. So that same template we will try to follow. But when we get to the linear system stage we will see that the questions are very different here.

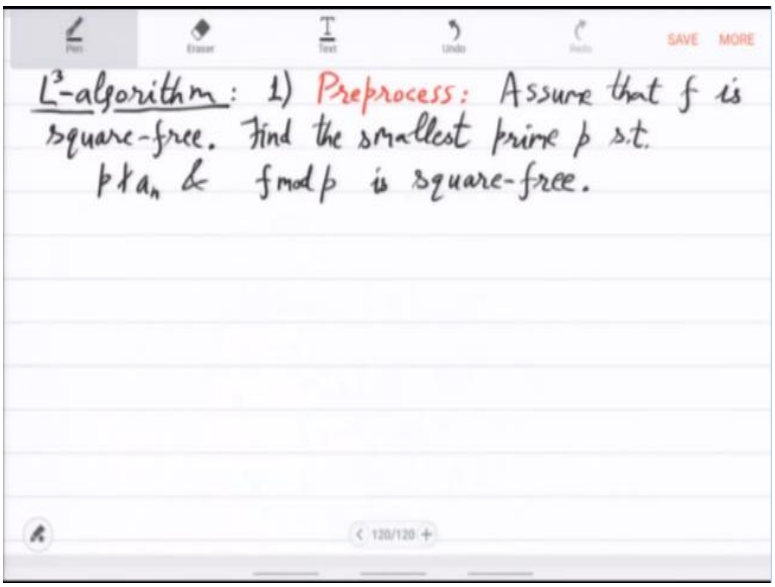
Okay, the questions are to do with integers. And a new question will appear which will take us into a new theory. Okay, so we will develop that theory. It is called the theory of lattices. And it has very interesting applications in also other areas. So let us implement this idea and see where we get stuck. So this new analysis was discovered by Hendrik Lenstra, RN Lenstra and Lovasz in 1982 igniting a new field.

Okay, so let us see the steps in slightly more detail. So input is f integral polynomial. Coefficients are not too big, right. So the coefficients are l bits, degree is n. What do you want in the output? Well, you want factors. So a non-trivial integral factor if it

exists, if it does not exist you should say that f is irreducible. Note that over complex f will always factor into linear factors.

But over rationals it may not, right? Over rationals for example, x square minus 2 is irreducible. So irreducibility testing is a non-trivial exercise, okay. So let us now go into L cube algorithm.

(Refer Slide Time: 47:35)



This algorithm is called L cube. The first step is preprocessing. Since you want to do Hensel lifting you want to factor f mod p and then Hensel lift. You want f mod p to have coprime factors. So for that, so that we have to ensure that. So you can first assume that f is square-free and which is easy to ensure because you can just take the GCD of f with f prime and if f was square-full you will already factor f.

So you can assume f to be square-free. But even if it is square-free mod p it may not be square-free, right. So when is it square-free mod p? So find the smallest prime p such that p should not divide a n and f mod p should be square-free, should remain square free. So p should not divide a n because we want the degree to remain the same. We do not want f degree to fall.