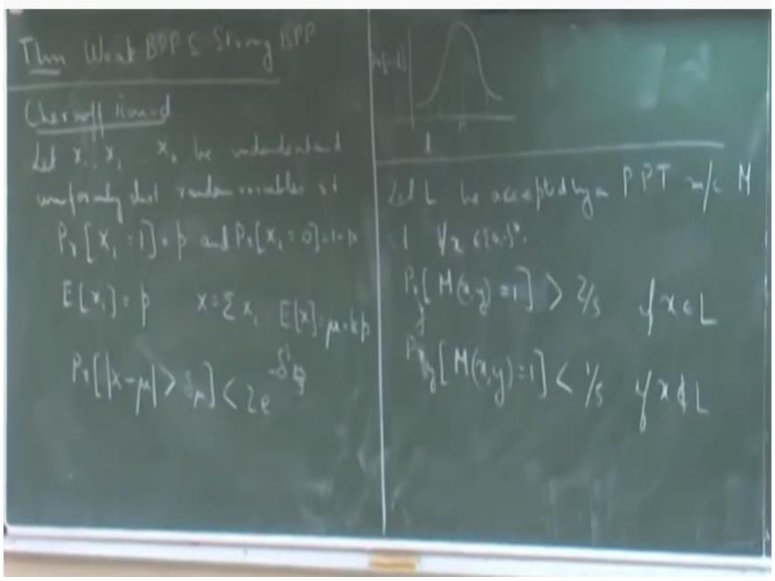
Computational Complexity Theory

Prof. Raghunath Tewari
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

Module No # 05
Lecture No # 21
Error Reduction of BPP

So good morning everybody so we will continue our discussion on probabilistic machines. So the first thing that we will see today is a claim that we left in our last class. So we saw these 2 variations in the definition of BPP we saw strong BPP and weak BPP and we argued that basically they are the same class they contain the same set of languages. So let us see why that claim is true.

(Refer Slide Time: 00:54)


Thm Weak BPP ⊆ Strong BPP
Chernoff Bound
Let $X_1, X_2 \ldots X_k$ be independent
uniformly dist random variables s.t
$\Pr[X_i = 1] = p$ and $\Pr[X_i = 0] = 1-p$
$E[X_i] = p$ $X = \sum X_i$ $E[X] = \mu = kp$
$\Pr[|X-\mu| > \delta\mu] < 2e^{-\delta^2\mu/3}$
Let L be accepted by a PPT m/c M
s.t $\forall x \in \{0,1\}^*$
$\Pr_y[M(x,y)=1] > 2/3$ if $x \in L$
$\Pr_y[M(x,y)=1] < 1/3$ if $x \notin L$

So in particular weak BPP is contained in strong BPP because the other direction is trivial it follows from the definition. So what I will do today to illustrate this proof is? I would not consider a general polynomial SN and PN I will just pick the some constant polynomial but the essentially idea will be illustrated in whatever we do. So let us get started so before getting into this proof let me talk a little bit about probability.

So there is something known as Chernoff bound in probability which bounds the tale of a distribution. So what so we mean by tale of distributions? So I will just come to that so let us see what Chernoff bound says? So suppose we have a random variables so let X 1, X 2 up to X k the independent and uniformly distributed random variables such that. So this condition is very important such that probability that X i = 1 is some constant p.

So p is some fixed number and probability X i takes the value 0 is 1 – p. So the important constraint on this random variable are they are they can take only 2 values. So they can be either 0 or 1 and each of those random variables they take 0 and 1 with the same probability. Any question so that is the condition that Chernoff of bound I mean under which whatever we claim next will be true. And for all i's the X i's take value 0 with probability 1 – p.

So you can think it off like coin tosses suppose you have a biased coin which gives you ahead with probability let us say 0.4 and tails with probability with 0.6. So if you make 10 tosses of the coin well at each step your probability of head is 0.4 and probability of tail is 0.6. And each of these tosses, are independent events and their uniformly distributed. So that is one picture you can keep in mind. And let so what is the expected value of an event X i?

Right, so what does it come out to be? It comes out to be p because it one times p + 0 times 1 – p. So therefore if I define x to be the sum of all these individual variables what is the expected value of x? Kp because expectations sums up so let us call this as Mu so expectation of x I will just denote this quantity as Mu. So then what Chernoff bound says is that the probability that x is bounded away from Mu that is the expected value of what x is?

By an amount of equal to sum delta times Mu is not more than 2 to the power 2 times e to the power minus delta square Mu by 3. So in other words the more deviate from Mu I mean the probability that we are deviating from Mu becomes exponentially smaller. So one way to look at this is you can think of this as so we think of this as graph so these random variables give as a bell shape curve here.

Where on y axis we have that probability that x takes the value let me not use t, here. So let me use something like l may be and probability that x takes the value l. And we have Mu sitting somewhere over here so the more we deviate from Mu the probability of that keeps decreasing exponentially. So that is what Chernoff bound states so there are, so this is what is called bounding the tale of the distribution.

So this is the distribution of the random variable x and these are the 2 tales of it where it is trailing down and I want to bound that what is the probability that random variable takes this value? What is the probability that it takes this value and thinks like that? And Chernoff bound the good thing about Chernoff bound is it gives an exponential bound on that tale. So there are other types of such bounds as well I do not know if you have heard of it like Markov's bound and Chebyshev's inequality.

So they also give a bound on the tale but for example in the case of Markov's inequality it only gives an inverse linear bound on the tale of the distribution and the reason for that is that it does not assume any property of the distribution. I mean x can be any random variable and it gives an inverse linear in the case of Chebyshev's inequality it assumes that the standard deviation is given and with that it gives an inverse quadratic bound.

But this gives a much stronger bound anyway so that is the no in our case because these value else are discrete values so. But you can think in it off is continuous also I mean in our case it is discrete the way we have defined this thing but we can also define probability in a continuous manner and we can have a continuous analog of this entire thing but. So that was the deviation from the theorem.

So let us get back to it so what we want to show is that we BPP is contained in strong BPP. So let l be accepted by probabilistic polynomial time somebody there so l is accepted by a probabilistic polynomial time machine m. Such that for all x let us say that probability that given random string y m of x, y accepts. Or let me write it as m of x, y = 1 is the same thing is let us say greater than 2 fifth and probability that one second let us write it here.

If x belong to l and the probability that m of x, y = 1 is less than let us say 1 fifth if x does not belong to l. So in particular and choosing my polynomial s n to be 5 and also the polynomial p n to be 5 in this case. But it does not matter so suppose if you toss a coin 10 times what is the probability that you will have let us say only one head and what do you think? Or let us say you have 0 heads so you can imagine that it will be something very low.

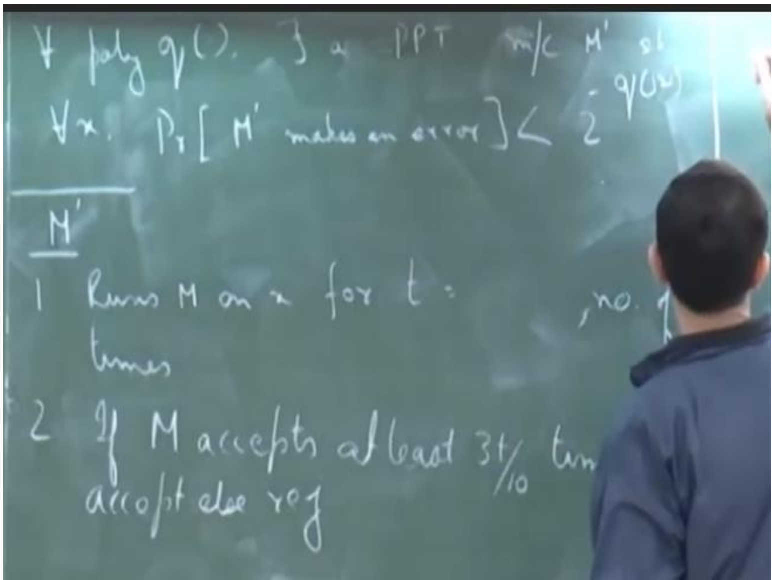
So binomial distribution will give you and similarly you can ask the question that what is the probability that you have only 1 head 2 head 3 head 5 head and so on? Up to 10 head so now let us say I want to so what is the expected number of heads that you get out of 10 coin tosses that is 5. So now if you want to ask the question that what is the probability that the number of heads is less than, 3. So that is it is at a distance of 2 away from the expected value of the total number of heads.

So that probability is basically given by Chernoff bound so Chernoff bound says that if, I want to calculate the probability that the number of heads is less than 3 or maybe greater than 7. In this case that is equal to 2 to the power 2 times e to the power minus whatever is that appropriate delta square times Mu by 3. So it gives you way of answering these questions. So instead of 10 maybe I am looking at million coin tosses and I ask that what is the probability that only 1 tenth of those fractions are heads.

So this thing will answer that question for you this is dependent on this distribution. So the distribution comes from the fact that you are doing coin tosses. NCK versus K it will be some kind of a exponential graph alright so NCK attains at maximum value for N by 2 for k = n by 2. And it is a exponential function so but no but one thing here is that it is only giving a an upper bound on the value that we have here need not be exact.

But still so if you plot a continuous curve then it will have something of this shape in NCK against K.

(Refer Slide Time: 15:11)



$\forall$ poly $q()$, $\exists$ a PPT m/c M' st

$\forall x$, $\Pr[$ M' makes an error $] < 2^{-q(|x|)}$

M'

1 Runs M on x for t = , no.
times

2 If M accepts at least $3t/10$ tim
accept else rej

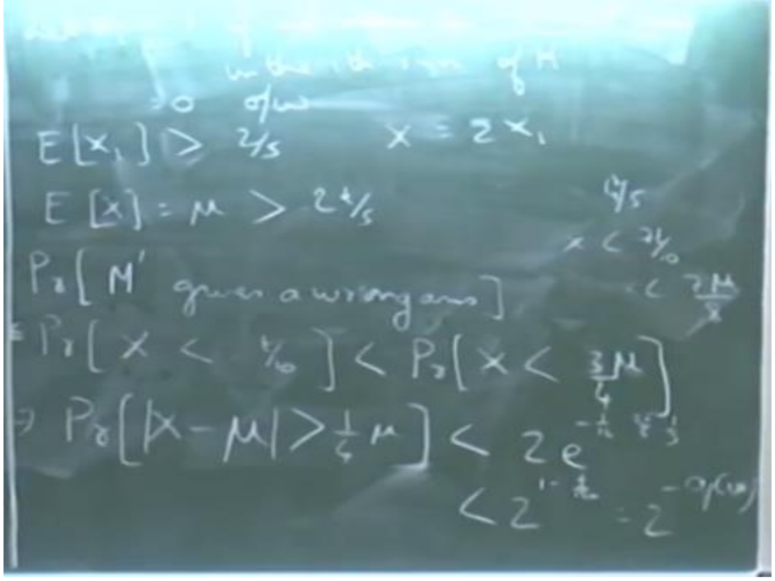
So we have this and then we want to show is for all polynomials q there exist a machine it has to be a polynomial time machine M prime such that. I will write it short such that for all x the probability that M prime makes an error is less than 2 to the power –q of x. I mean I just shortened these statements by saying that there are probability of M prime is at most 2 to the power –q of x.

So let us see how to design M prime? So what M prime does is actually quite simple so it does so first it runs M on x for t is equal to so we will just leave this as blank for the time being. So many number of times so we, will just think of this as t for the time being and then if at least 3 t by 10 of. Then if M accepts at least 3 t by 10 times the, accept else reject so let us understand this algorithm M prime.

So what it does is that given an input x it just runs the machine M on this x for t number of times. So at each time with a certain probability the machine either accepts or rejects. And then finally if M accepts at least 3 t by 10 times out of, these t many runs then we accept else we reject. So the reason why we pick this number 3 t by 10 is because it lies between 1 fifth and 2 fifth. So that is the logic behind it so now let us see that what is the probability that M times make an error?

So here where we leaves Chernoff bound. So before going into that let us define something.

(Refer Slide Time: 19:18)


in the i th run of M
0 o/w
$E[X_1] > 2/5$
$X = \Sigma X_i$
$E[X] = \mu > 2t/5$
$4/5$
$X < 3t/10$
$< \frac{3\mu}{4}$
$Pr[M'$ gives a wrong ans$]$
$= Pr[X < 3t/10] < Pr[X < \frac{3\mu}{4}]$
$\Rightarrow Pr[|X-\mu| > \frac{1}{4}\mu] < 2e^{-\frac{1}{16}\cdot\frac{t}{3}}$
$< 2^{1-\frac{t}{48}} = 2^{-q(|x|)}$

So let X i be the random variable which is equal to 1 if M prime gives the correct answer in the ith run. And let us say it is equal to 0 otherwise so note that this is again a random variable which takes only 2 values. Because at the ith run m prime so when I am running M on x for the ith time so M prime either gives the correct answer or it gives the wrong answer M. So in the ith run of M prime here I will just say it as M gives the correct answer in the ith run of I mean.

So just so maybe I will just frame the sentence as if so if we obtain the correct answer in the ith run of M. So that is when we fix this as 1 and otherwise 0 so what is the expected value of X i. So suppose we have a string which is in the language what is the expected value of X i in that case greater than 2 by 5. And in the other case when x does not belong to l no think about it.

So if x is does not belong to l what is the probability that M x, y rejects it is greater than 4 by 5. So here the thing is that X i = 1 if we obtain the correct answer so if the string is in the language it says accept and if the string is not in the language it says rejects. So we can give an upper bound on, E of x y as always greater than 2 by 5. So in case it is greater than 1 by 5 and in the other case it is greater than 4 by 5.

But this is anyway lower bound so therefore if we consider the random variable x which is again the sum of the variables X i the expected value of x. So again let me denote this by Mu so, this is greater than equal to actually these are strict in equalities. I mean the way I define them so just keep that thing preserved so that is 2 t by 5. So now we are in a position to apply Chernoff’s bound so well not quite so let me get there so then what is the probability that M prime gives a wrong answer.

So when is M prime giving a wrong answer think for a while so let us look at take one case suppose we have a string which is in the language. So then M prime will accept if at least 3 t by 10 times M accepts otherwise it rejects. So that means M prime giving a wrong answer when X is less than 3 t by 10 correct? Because if X is greater than 3 t by 10 I mean the way I define the X is it will anyway give you the right answer.

So this again if I substitute t for Mu so this is the probability that X is smaller than what? This is by simple substitution because Mu is greater than 2 t by 5 so I just substituted Mu instead of t. So which implies that the probability that X – Mu is greater than 1 fourth of Mu is whatever

Chernoff bound gives us. So how are we getting this probability from this so if X is less than 3 fourth of Mu what it means is that the difference between X and Mu is at least 1 fourth of Mu.

This actually this probability is a slightly more, higher because this is true if X is less than 3 fourth of Mu as well as if X is greater than 5 fourth of Mu but anyway it does not bother us. Because we are only trying to give a upper bound and now we can apply Chernoff bound here. So this is nothing but 2 times e raised to what is our delta? So we have 1 over 16 times what is our Mu? Mu is greater than 2 t by 5 so we have 2 t by 5 times 1 third and if you just work this out.

This is less than 2 to the power 1- there should be a minus here 1- t over 120. So now we can go back and fix what is that t that we want it is 120 time. So we want the error to be bounded by q of x so it is 120 times it is because of this 2 I mean I can just write I mean I can just change e to 2. Because e is anyway greater than 2 and that is extra one so if I substitute this particular t over here what we get is? This is equal to 2 to the power -q of x.

Suppose it belongs to the language I get a wrong answer when I only have less than 3 t by 10 accepting answers for n and actually the same thing also happens is x does not belong to l. So in that case I would so if x does not below to l. So that is true but I am just saying that why is this probability the same as this event. So the probability that M prime is giving a wrong answer so if X is in l or random variable is less than 3 t by 10 and if X does not belong to l.

So what do we have here? So exactly so I mean basically the way this X as defined so see X i = 1 if we are obtaining the correct answer. In other words if the machine I mean if the string is not in the language then X i will be equal to one only if this machine rejects for at least so many number of times. And that probability is again greater than 4 fifth because of the way we have our machine M and that anyway gives you a lower bound of 2 t by 5 on the expected value of X.

Yes it is rejecting for 70 by 10 times and for those 70 by 10 times you are getting a 1 here where here no M dash is giving a wrong answer. So it is accepting less than 3 t by 10, number of times. It is for correct answer correct no see how is X is defined so it is rejecting 70 by 10 times at least. But for each of those reject you are getting a 1 here. So the probability; that you are getting a wrong answer which is accepting.

So what, is the probability that you are getting wrong answer that is M prime accepts so that is now less than 3 t by 10.

“Professor - Student conversation starts”
We are getting 1 in the case M rejects and if X does not belongs to 1 for X i and it rejects at least 70 by 10 time X value will be at least 70 by 10 because X is the summation of all X.
“Professor - Student conversation ends”

So what you are saying is just say that again correct at least 70 by 10 so maybe I should replace this with a less than equal to. Because in one case then it should be less than 3 t by 10 and the other case 70 by 10 so anyway this is a smaller probability I mean anyway this is a larger probability. So if we take this less than 70 by 10 anyway so anyway it does not give a wrong answer only you have a different value of t.

But let me just see the definition of M prime so no so if M accepts at least 3 t by 10 times then we accept. So if we are rejecting it means that it is accepting less than 3 t by 10 times which is more than 70 by 10 as we correctly pointed out. So if it is rejecting more than 70 by 10 times. Here the sum of these random variables is so here basically I have that X is less than 70 by 10 in that case.

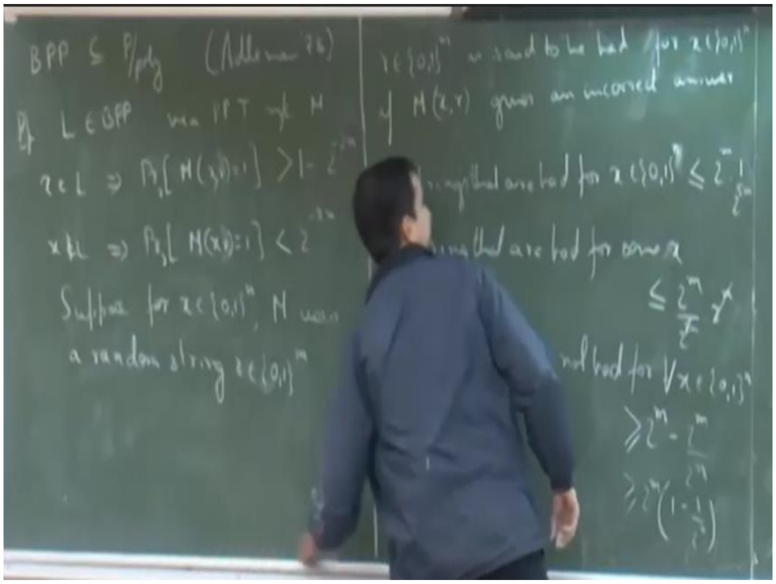
So then we can appropriately change all this things so what do we get? So we have 70 by 10 here so that will give us no there is something wrong here. Because that does not make sense so in case we have that the probability is less than 3 t by 10 and; in the other case it is less than 70 by 10 that you know both are equidistant from Mu 3 t by 10 and 70 by 10. 4 t by 10 but you know that you are getting by Chernoff bound but how can you say that here.

Anyway so let us see so in this case so till this point I think it is fine instead of writing this as M prime gives wrong answer what a? So one case is that M prime rejects when X belongs to l so that is the probability that we are bounding here actually. On the other side we have to consider that M prime accepts when X does not belong to l. So then we have Mu as 40 by 5 and we have X is less than 70 by 10 so therefore this is X is less than 7 Mu by 8.

So then I think it will work out fine and then finally the polynomial that you can take is the max of those 2 polynomials. So this will give you a polynomial q and the other one is giving some other polynomial q prime and you can take the max of those 2. So I think that makes sense also I mean I kind of overlook this but thanks for pointing it out make sense. Because we should have different bound on the 2 different cases so is that clear to everybody?

So it is basically just running through the same sequence of analysis but taking the value of Mu appropriate. I think the other one will be slightly larger but anyway so let us not bother about that. So what basically this shows is that it does not matter what is the error bound that we consider for BPP. And in particular we can go as close to inverse polynomial so the next result that we will see again regarding BPP is that.

(Refer Slide Time: 37:39)



BPP ⊆ P/poly (Adleman 78)
Pf L ∈ BPP via PPT M
x ∈ L ⇒ Pr[M(x,r)=1] > 1 - 2^{-2n}
x ∉ L ⇒ Pr[M(x,r)=1] < 2^{-2n}
Suppose for x ∈ {0,1}^n, M uses a random string r ∈ {0,1}^m
r ∈ {0,1}^m is said to be bad for x ∈ {0,1}^n if M(x,r) gives an incorrect answer
strings that are bad for x ∈ {0,1}^n ≤ 2^m · 1/2^{2n}
strings that are bad for some x ≤ 2^m/2^n
not bad for ∀x ∈ {0,1}^n ≥ 2^m - 2^m/2^n ≥ 2^m(1 - 1/2^n)

BPP is contained in P by poly and this was shown by Adleman in 78 so this follows from a very simple counting argument. So suppose we have a language l in BPP then by the definition of strong BPP we can assume that X belongs to l then the probability that M so let us say l in BPP via a probabilistic polynomial time machine M probability that M just write this as. Let us say we have it as greater than 2 to the power 1 – 1 to the power -2 n.

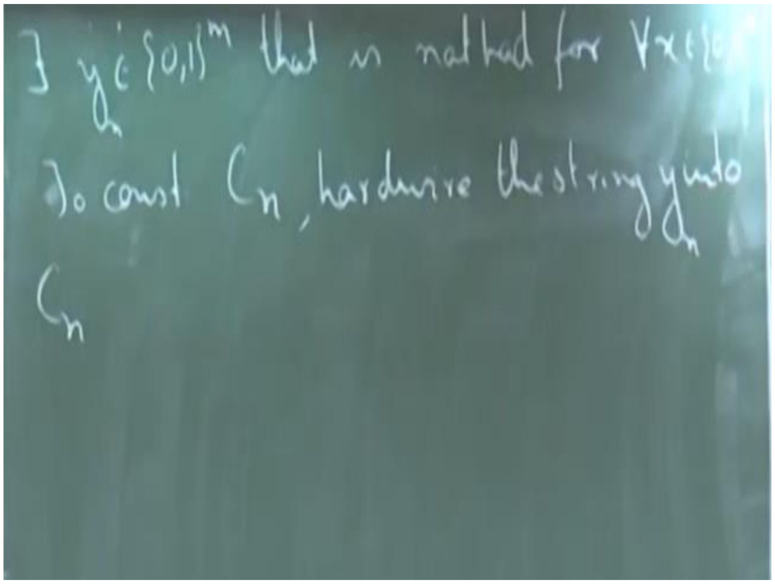
And if X does not belong to l then the probability that M accepts is less than 2 to the power -2 n. So we are just fixing this q to be 2 n so now suppose that for a string X of length n M uses random string of M bits. So what we essentially want to count is so let us define a random string as bad. So is said to be bad for some x in 0, 1 to the power n if M on x, r gives an incorrect answer.

So the, number of strings that are bad for x so let us say we fix an x so how many strings are there that are bad for a fixed x. In other words that will give you incorrect answer maybe I can just write this as probability over r and M of x, r = 1. Just to illustrate what we mean by r here. So how many strings are bad for a fixed x? It is at most 2 to the power n so that is the total number of strings and we have a 1 over 2 to the power 2 n fraction of then that are bad from this probability.

So therefore the number of strings that are bad for some x so in other words I am calling a strings bad for some x if there is some x in 0, 1 to the power n for which that string is giving an incorrect answer. So how much is that? So how many such strings do we have? In 0, 1 to the power n, 2 to the power n so we can just simply apply the union bound. So we have 2 to the power n divided by 2 to the power 2 n times 2 t to the power n so these 2 cancels off.

So we have so 2 to the power M by 2 to the power n strings that are bad for some x. So therefore the number of strings I am just taking the complement of this event now that are not bad for all x in 0, 1 to the power n is how much? So I am just taking the complement of this set that is greater than. So the total number of such strings; are 2 to the power n so it is 2 to the power n – 2 to the power n by 2 to the power n. Which is 2 to the power n times 1 – 1 over 2 to the power n so what does is this tell us?

(Refer Slide Time: 45:09)



$\exists\ y_n \in \{0,1\}^m$ that is not bad for $\forall x \in \{0$

To const C_n, hardwire the string y_n into

C_n

That there exists; a y 0, 1 to the power m that is not bad for all x in 0, 1 to the power n. So what was the definition of bad? So we said that the string was bad if it gave an incorrect answer so what we have shown here is that? There is some y which is actually good for all x so then what we can do is so to construct the circuit C of n just hardwire a string y into C of n. So every n a just hardwire that appropriate y so we can just call this as y of n to emphasize that it is for that particular n.

Into that circuit and go ahead with the polynomial time computation so again this only gives an existential result. It only shows that there exist a; circuit family which can compute BPP so a pretty straight forward just doing some counting argument. So we will stop here today and will carry on more about these things on Wednesday