CSA E0 235: Cryptography **26 Aug 2019**

Scribe for Lecture 6

Instructor: Arpita Patra *Submitted by: Patil Abhijit Suresh*

1 Recall

In the previous lecture we have introduced Pseudo-randomness and Pseudo-random Generators (PRGs). We have given indistinguishability based definition of PRG security and saw Statistical test and Next bit security test for PRG. We also understood that PRGs are **not secure against unbounded adversary** and the first important assumption of Course i.e **"PRG exists"**

We gave indistinguishability based definition for COA attack and proved a COA secure SKE under the assumptions of existence of PRG along with shortcomings of the current construction/definition.

We introduced the concept of Multiple Message COA and understood that Multiple Message COA security is *stronger* notion than Single message COA security and thus requires encryption algorithm to be randomized.

Lecture concluded with brief introduction to applications of PRG and ind secure SKE such as Roulette Games and Anonymous Message Transfer/Onion Routing.

2 Today's Goal

In this lecture we will discuss more about the following :

- Introduction to Hybrid Arguments
- PRG with one bit expansion implies PRG with polynomial expansion and its proof using hybrid arguments
- Application of PRG
  - Coin Tossing
  - Commitment Schemes

3 Introduction to Hybrid Arguments

A hybrid argument is a proof technique used to prove indistinguishability. It is used when a basic primitive is applied multiple times. There exists Hybrid Arguments which uses several different primitives and apply them multiple times however in this course we will only be discussing about Hybrids which uses single primitive.

The technique works by defining a series of intermediate distributions that bridge between two extreme distributions that we wish to prove indistinguishable.

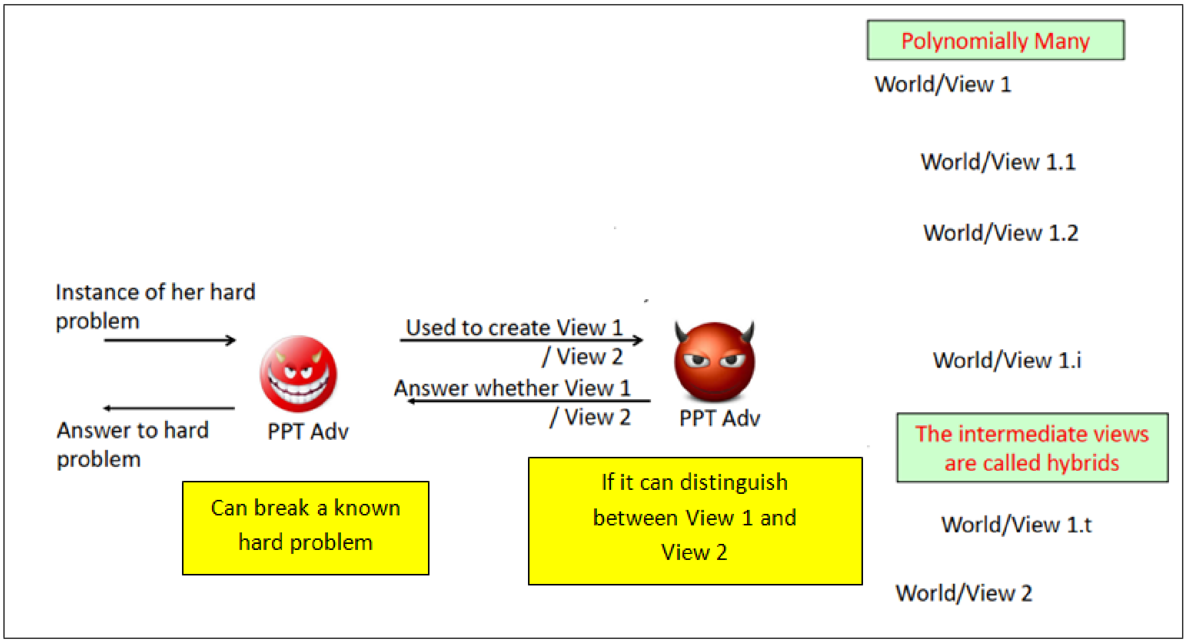

Polynomially Many
World/View 1
World/View 1.1
World/View 1.2
Instance of her hard problem
Used to create View 1 / View 2
Answer whether View 1 / View 2
PPT Adv
PPT Adv
World/View 1.i
Answer to hard problem
The intermediate views are called hybrids
Can break a known hard problem
If it can distinguish between View 1 and View 2
World/View 1.t
World/View 2

Figure 1: Hybrid Arguments.

The extreme distribution are also called as View 1/ View 2 , Left Hybrid/Right Hybrid and Left View/Right View. Sometimes one instance of hard problem is not sufficient to create these View 1 and View 2. So we define series of intermediate view starting from View 1 and will gradually move towards view 2. These intermediate views are called as Hybrids.

To apply Hybrid Proof techniques, following three conditions should hold

1. Extreme Distributions should be the one for which indistinguishability need to be proven
2. Most important is that it must be possible to translate the capability of distinguishing consecutive hybrid distributions into breaking some underlying assumption. This in turn translate into relation between extreme views.
3. Finally the number of Hybrid must be Polynomial

Let us see How Hybrid Arguments work mathematically :-

Consider View 1 and View 2 and we have to prove indistinguishability between these two views. From view 1 after minor adjustment we can move to View 1.1 and then try to prove View 1 and View 1.1 are indistinguishable

$$|Pr[A(View1) = 1] – Pr[A(View1.1) = 1]| < negl(n)$$

Similarly from View 1.1 we can obtain View 1.2 and so on......

$$|Pr[A(View1.1) = 1] – Pr[A(View1.2) = 1]| < negl(n)$$

$$|Pr[A(View1.i)] = 1 – Pr[A(View1.t) = 1]| < negl(n)$$

At the end after polynomial such iteration we reach to View 2.

$$|Pr[A(View1.t) = 1] – Pr[A(View2) = 1]| < negl(n)$$

Since there are t (polynomial no) such expression adding all of the above we get

$$|Pr[A(View1) = 1] – Pr[A(View2) = 1]| < t.negl(n) < negl(n')$$

Hence View 1 and View 2 can be proved to indistinguishable.

4 PRG with Polynomial Expansion Factor

Theorem: If there is a PRG with expansion factor l(n) = n+1, then for any poly(n), there exists a PRG G' with expansion factor poly(n) i.e

$$\{PRG[G : \{0,1\}^n \to \{0,1\}^n + 1]\} \Rightarrow \{PRG[G' : \{0,1\}^n \to \{0,1\}^{poly(n)}]\}$$

Given G we have to build G'. First let us try to get n+2 bits from n+1 bits.

Given an initial seed s, it computes t1 = G(s) to obtain n + 1 pseudo-random bits. The initial n bits of t1 are then used again as a seed for G; the resulting n+1 bits, concatenated with the final bit of t1, yield the (n + 2) bit output

Similarly extending the arguments, We can get n+p(n) bits

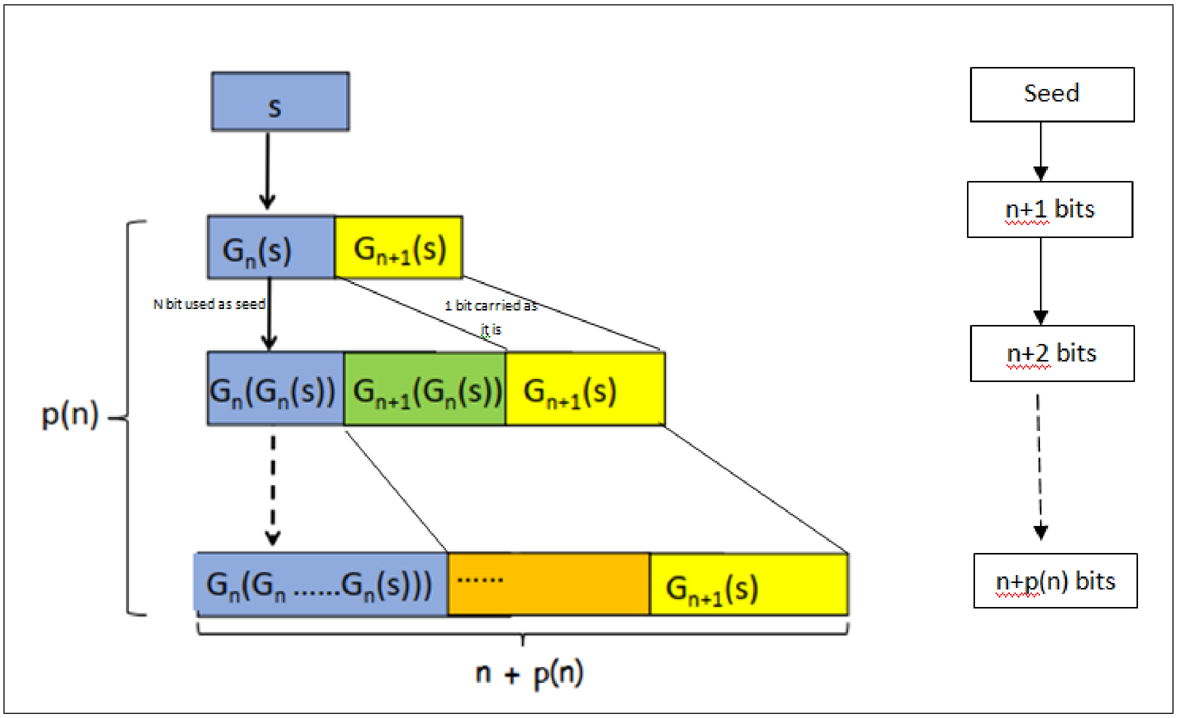


Figure 2: Increasing the expansion of a pseudo-random generator

4.1 Proof via Hybrid Argument

4.1.1 Motivation for Hybrid Arguments

Informally we are trying to prove if G is a PRG then G' is a PRG. From distinguisher of G we are trying to make distinguisher for G'. However designing challenge for G' from G will be difficult. Hence proof by reduction will not work.

4.1.2 The Proof

Notations

View 1 H_0 : Distribution on leaves when the root (0th level node) is a random string i.e all the bits are Pseudo-random

View 2 $\mathrm{H}_{\mathrm{n+p(n)}}$: Distributions on leaves when the leaves (p(n)th level nodes) are random strings i.e. Truly Random RS

Intermediate Hybrid

Considering the extreme Views and to move from View 1 to View 2 we can take intermediate hybrid as View 1.1 H_1 : Distribution on leaves when the root (1st level node) is a random string i.e RS

View 1.2 H_2 : Distribution on leaves when the root (2nd level node) is a random string i.e RS

.... and so on

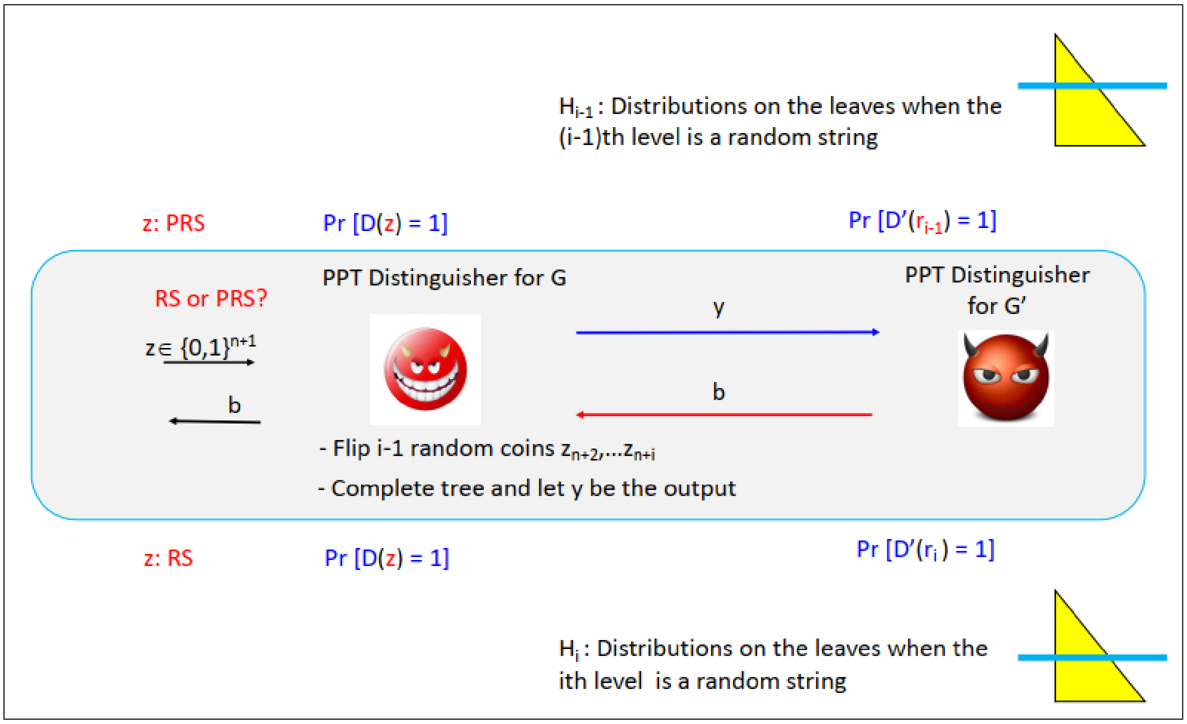
Let us try to prove indistinguishability between Hybrid i and Hybrid i-1 where D is a polynomial time distinguisher

To Prove

$$|Pr[D(r_{\text{i-1}}) = 1] – Pr[D(r_{\text{i}})]| < negl(n)$$

Proof Given G is a PRG, we know that PPT distinguisher D can distinguish between Random String(RS) and Pseudo-random String(PRS) with negligible property

$$|Pr[D(G(s)) = 1] – Pr[D(r)]| < negl(n)$$



H_{i-1} : Distributions on the leaves when the (i-1)th level is a random string
z: PRS
Pr [D(z) = 1]
Pr [D'(r_{i-1}) = 1]
RS or PRS?
PPT Distinguisher for G
PPT Distinguisher for G'
z∈ {0,1}^{n+1}
y
b
b
- Flip i-1 random coins z_{n+2},...z_{n+i}
- Complete tree and let y be the output
z: RS
Pr [D(z) = 1]
Pr [D'(r_i) = 1]
H_i : Distributions on the leaves when the ith level is a random string

Figure 3: indistinguishability between Hybrid i and Hybrid i-1

Consider Distribution H_{i-1} and H_i

H_{i-1} level has last i-1 bits as Random string and rest all bits from(0 to n-i-1 bits as) pseudo random.

Similarly H_i level has last i bits as Random String and rest all bits from (o to n+i) as Pseudo Random. Last i-1 bit in both are random as they are obtained Adversary G by flipping Coin.

Hence The i th bit from end in H_{i-1} is Pseudo-random and in H_i is random.

$$H_{i\text{-}1} = PRS$$

$$H_i = RS$$

Hence Consider reduction in which PPT distinguisher G takes n+1 bit which it want to distinguish as RS or PRS He translate it into G' as follows:- He flips i-1 coins, appends it to the string given by challenger and generate a string n+i bit

If G is PRS then generated distribution is same as H_{i-1}. Last i-1bits coming from coin toss as random and last ith bit coming from G as pseudo random.

$$Pr[D(G(s)) = 1] = Pr[D(r_{i\text{-}1}) = 1]$$

However if G is RS then generated distribution is same as H_i. Last i-1 bits coming from coin toss as random and last i th bit coming from G is also random.

$$Pr[D(r) = 1] = Pr[D(r_i)]$$

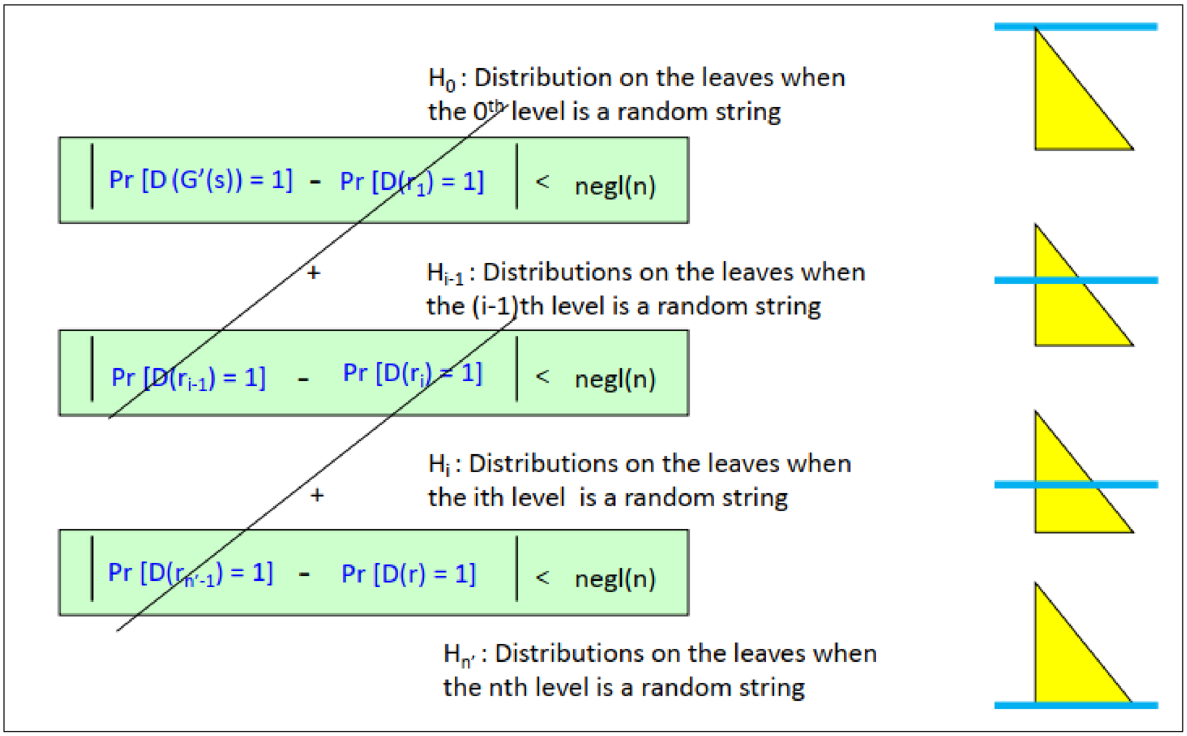
But We know that G is PRG

$$|Pr[D(G(s)) = 1] – Pr[D(r) = 1]| < negl(n)$$

Hence

$$|Pr[D(r_{\text{i-1}}) = 1] – Pr[D(r_{\text{i}})]| < negl(n)$$

Concluding the Hybrid Argument and summing it over all intermediate views we get



H_0 : Distribution on the leaves when the 0th level is a random string
| Pr [D (G'(s)) = 1] - Pr [D(r_1) = 1] | < negl(n)
+
H_{i-1} : Distributions on the leaves when the (i-1)th level is a random string
| Pr [D(r_{i-1}) = 1] - Pr [D(r_i) = 1] | < negl(n)
H_i : Distributions on the leaves when the ith level is a random string
+
| Pr [D($r_{n'-1}$) = 1] - Pr [D(r) = 1] | < negl(n)
$H_{n'}$: Distributions on the leaves when the nth level is a random string

Figure 4: Summation over all intermediate Hybrid

$$|Pr[D(r_0) = 1] – Pr[D(r_{\text{n+p(n)}}) = 1]| < negl(n)$$

$$|Pr[D(G'(s)) = 1] – Pr[D(r) = 1]| < negl(n)$$

Hence View 1 and View 2 are indistinguishable.
Therefore G'(s) is a PRG given G is PRG.

In another Word If there is a PRG with expansion factor l(n) = n+1, then for any poly(n), there exists a PRG G' with expansion factor poly(n).

■

5 Applications of PRG - Coin-flipping and Commitments

Goal:Two parties Sender and Receiver has to agree on an unbiased random bit.

The problem is of significant important and has many real world applications. The problem is trivial when both parties are co-located however when they are far apart it becomes harder and complex.

PRG gives us Commitment Scheme which can be used in Coin flipping application. Let us first understand the Coin Flipping Problem.

5.1 Coin Flipping Problem

Two Entities: A Sender S Saina and Receiver R Rajdeep wants to agree on unbiased bit.

S has a private b$_0$ which it wants to "commit" to R. S computes a "commitment" c of m and sends it to R. R sends bit b$_1$ to S. Later S Sends opening information to R and reveals an m by opening the commitment c with opening information. Both Output $b_0 \oplus b_1$. Using a commitment scheme, S and R can generate a random bit $b \in \{0, 1\}$ so that no side can bias the result towards their preferred outcome. Let us see how?

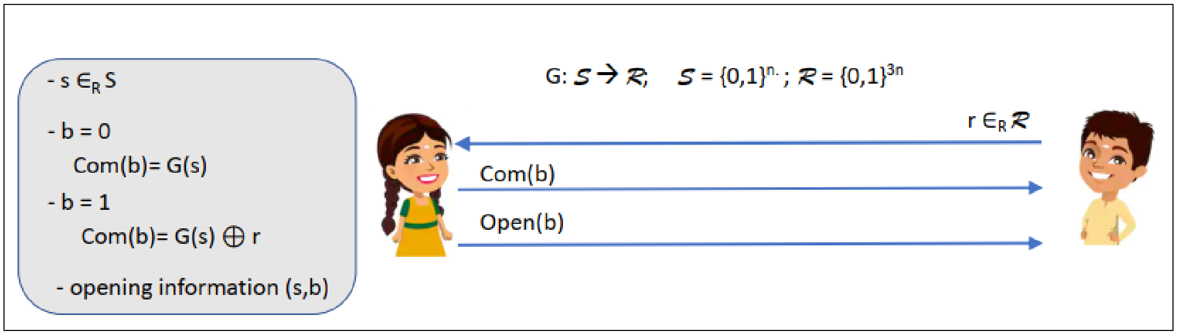


Figure 5: Bit-Commitment from PRG

5.2 Commitment Schemes

A bit commitment scheme can be construed using secure PRG. Consider the following scheme

Let G :$S \to R$ be a secure PRG where $|R| \geq |S|^3$ and $R = \{0, 1\}^n$ for some n. To commit to the bit b$_0$, S and R engage in the following protocol:-

Saina commits to bit $b_0 \in \{0, 1\}$:

Step 1: Rajdeep chooses a random $r \in R$ and sends r to Saina.

Step 2: Saina chooses a random $s \in S$ and computes $c \leftarrow (s, r, b0)$ where com(s, r, b0) is the following function:

$$c = com(s, r, b0) = \begin{cases} G(s) & if b = 0 \\ G(s) \oplus r & if b = 1 \end{cases}$$

When it comes time to open the commitment Saina sends (b0, s) to Rajdeep. Rajdeep accepts the opening if $c = com(s, r, b0)$ and rejects otherwise

5.3 Security of Commitment Schemes and its Proof

A commitment scheme is secure if it satisfies the following two properties:

- **Hiding** : The commitment string c reveals no information about the committed bit b. More precisely, the distribution on c when committing to the bit 0 is indistinguishable from the distribution on c when committing to the bit 1.

Proof:-

The hiding property follows directly from the security of the PRG: because the output G(s) is computationally indistinguishable from a uniform random string in R it follows that $G(s) \oplus r$ is also computationally indistinguishable from a uniform random string in R. Therefore, whether b$_0$ = 0 or b$_0$ = 1, the commitment string c is computationally indistinguishable from a uniform string in R, as required.

However as we know PRG is only secure against PPT adversary the property is true only for PPT adversary.

- **Binding**: This ensures that once Saina commits to a bit b she can open it as b and nothing else. In the commitment scheme we present the binding property holds unconditionally

Proof:-

The only way Saina can open a commitment b as both 0 and 1 is if there exist two seeds $s_0, s_1 \in S$ such that

$$c = G(s_0) = G(s_1) \oplus r$$

$$\Rightarrow G(s_0) \oplus G(s_1) = r$$

Let us say that $r \in R$ is "bad" if there are seeds $s_0, s_1 \in S$ such that $G(s0) \oplus G(s1) = r$.

Total No of pair of seeds $\#(s_0, s_1) = \{0,1\}^{2n} = |S|^2$

$\#BadR's = |S|^2$

Also According to Commitment Protocol $|R| \geq |S|^3$

Probability of Choosing Bad R

$$P(Bad\,R's) = \frac{\#Bad\,R's}{Total\,R} = \frac{|S|^2}{|R|} = \frac{2^{2n}}{2^{3n}} = \frac{1}{2^n} = negl(n)$$

Therefore, the probability that Sender can open the commitment c as both 0 and 1 is negligible.

Considering both Hiding and Binding property we can say that if one party is honest, other party can not bias the output.

- **Corrupt Rajdeep**: Rajdeep's pick is independent of Saina's pick (which is random) and so outcome is random
- **Corrupt Saina**: Saina cannot change her committed bit upon seeing Rajdeep's bit (which is random) and so outcome is random

6 Conclusion

In today's class we introduced concept of hybrid proof and the proof of PRG with polynomial bit expansion. We have also seen application of PRG in coin flipping. We had planned to cover Introduction to CPA security in this class but due to paucity of time same will be covered in next class.

7 References

[1] Arpita Patra, *https://www.csa.iisc.ac.in/ cris/e0 235.html*

[2] Jonathan Katz and Yehuda Lindell, *Introduction to the modern cryptography*

[3] Dan Boneh and Victor Shoup, *A Graduate Course in Applied Cryptography*

[4] Internet,*https://www.crypto.stackexchange.com*

[5] Internet,*https://www.coursera.org/learn*