Disclaimer: *These notes have not been subjected to the usual scrutiny for formal publications. They are to be used only for the class.*
Usage: Here is a listing of topics modified from quiz 2 review topics (with later stuff added). This is meant more of a self-check questionnaire: see if you can answer the questions in the bullet lists.

For the final, you should review the lecture notes (LN), the homeworks (HW) and the recitation notes (RN), as well as handouts and referenced chapters in the textbook (M&R)

1. Balls & Bins - where did you see the following quantities? What do they remind you of? "$n \ln n$", "$\sqrt{n}$" (assume n *labeled* balls and n bins) ... (hint: coupon collecting, (generalized) birthday problem, LN #1, RN #1)

2. PIE (principal of inclusion and exclusion)

   (a) how would you generalize the following equality to more than 2 events?

   $$Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)$$

   (... hint: see LN #3 for PIE)

   (b) recall that theorem 1' in LN #3 generalizes PIE

   (c) when do you say that a permutation is a "derangement"? how would you define it in terms of "cycle"s? (... hint: see LN #3 for the cycle representation of permutations see RN #4 for derangements)

   (d) recall the distribution of the number of cycles in a random permutation. what does the distribution look like when n goes to infinity? (... hint: see LN #3)

   (e) what is the Boole-Bonferroni inequality? (... hint: we used it in HW #3, P6(e))

3. Probability basics:

   (a) sample space, events, random variables

      i. what do you think is the difference between a random variable and an ordinary variable? ... (hint: a r.v. is more like a function)

   (b) independence:

      i. how is conditional independence defined?

      ii. (ADDED) Recall Markov's property, which is a form of conditional independence, that is,

      $$\Pr[X_i \mid X_{i-1}, .., X_0] = \Pr[X_i \mid X_{i-1}] \qquad \text{or,} \qquad E[X_i \mid X_{i-1}, .., X_0] = E[X_i \mid X_{i-1}]$$

      iii. what is the difference between pairwise and mutual independence?

      iv. given two events A and B, what's the difference between saying that they are independent versus saying that they are mutually exclusive?

(c) distribution

i. recall joint/marginal/conditional distributions

ii. what is total probability theorem? where did you use it? recall the lightening problem in Quiz 1, and HW #3 Problem 1

iii. what is chain rule of probability? (recall that we used it in analysis of Karger's mincut algorithm: the place where we computed the survival probability of a min-cut)

(d) moments

i. recall the definition of expected value and variance

ii. recall linearity of expectation

iii. recall conditional expectation? what is double expectation theorem? (... hint: see RN #6, also compare with method of conditional expectation in derandomization, e.g., MAX-SAT)

iv. when two r.v. are independent, does it bring up any property of expected value or variance? (... hint: if two r.v. X and Y are indept, $E[XY] = ... = E[X]E[Y]$, and $\text{Var}[X+Y] = \text{Var}[X] + \text{Var}[Y]$)

v. recall(or look up) the definition of standard deviation and covariance

vi. (ADDED) What about conditional distribution? Since it is a probability distribution, we can define its moments: mean, variance, etc. (cf. the topic Conditional distribution)

(e) well-known distributions

i. what is the geometric distribution? (... hint: the number of coin tosses until the first head (and includes the first head))

ii. what is the binomial distribution?

iii. what is the Poisson distribution? review HW #3 P1 and P2 for properties.

iv. when does binomial distribution become like a Poisson distribution? (... hint: when $np = \lambda$ and n goes to infinity, see LN #3 towards the end)

v. what is the normal (or the Gaussian) distribution? when does binomial distribution become like a normal distribution? (... hint: when p=1/2 and n goes to infinity. recall central limit theorem, LN #6)

(f) (ADDED) Conditional distribution

i. Recall total probability theorem. And recall the double expectation theorem (a.k.a. "towering property"). Write them down. Do they look similar? Why? (hint: think about the general approach of "case analysis", that is, to obtain an unknown quantity, enumerate possible cases, and compute the unknown conditioned on each case)

ii. conditional expectation is linear (just like ordinary expectation) (hint: for proofs of the properties of conditional distribution, see RN#6)

(g) asymptotic theorems

i. CLT (central limit theorem) in the context of balancing lights (LN #6)

ii. WLLN (weak law of large numbers) in the context of optimal coding (LN #8)

4. Information theory:

   (a) when we code a series of numbers that are realizations of a random variable. how does the notion of entropy come into the play? the answer is broken into several steps (LN #8)

   what is the entropy of a discrete random variable? what is a "typical set"? what properties does it possess? (hint: ... it's small and it has almost all the probability)

   (b) why did we say it's only a proof of existence and not practical for optimal coding? (cf. Probabilistic methods topic) (hint: ... how would you find the typical set? then, how many codewords do you have to keep?)

   (c) recall Huffman coding, in what sense is it optimal? (hint: .. average codeword length)

5. Probability bounds:

   (a) what is Markov's inequality? what happens if the r.v. can take on negative values? when is the bound tight? (... hint: Markov's inequality is covered in LN #1, page 3; see also HW #5 P2)

   (b) what is Chebyshev's inequality? what happens if the r.v. has infinite variance? when is the bound tight? (... hint: Chebyshev's inequality was first encountered in LN #7, page 2, where we used it to bound the probability given the expectation; the inequality becomes vacuous when Var[X] diverges; you were asked to make the bound tight in HW #5, P2)

   (c) (ADDED) Chernoff bounds:

      i. recall the difference between additive and multiplicative Chernoff bounds

      ii. recall the three ingredients in proving various forms of the Chernoff bounds (hint: .. first, introduce the *moment generating function* $E[e^{tX}]$. Link expectation and tail probability using Markov's inequality. Exploit Independence to break the expectation into a product. Relax each term in some way. see RN #10 and M&R section 4.1 for details)

      iii. Given a sum of a number of random variables, what is the *key* condition for Chernoff bound to hold for the sum? (hint: .. independence!)

      iv. How can we use Chernoff bound to say anything about a random variable a long distance away from its mean? (hint: recall HW #9 P4)

   (d) (ADDED) Martingale and Azuma's inequality:

      i. Recall the definition of a martingale. (consult the hand out on martingale from Alon & Spencer's book)

      ii. What is a Doob martingale? (look up in M&R if necessary) This is a large class of martingales that cover a lot of applications of martingale.

      iii. Besides being a martingale, what's the other condition for Azuma's inequality to hold? (hint: .. bounded difference)

iv. Compare Chernoff and Azuma, which assumes less? (Chernoff assumes independence, Azuma requires a martingale, does one of them imply the other?) Recall HW #10, P1, where we used a martingale setup to reach the same conclusion from Chernoff bound, in particular, the random variables in a martingale does *not* have to be independent.

(e) comparison: Markov links probability with expectation, and the bound is inversely proportional to the deviation λ; Chebyshev links probability with variance and the bound is inversely quadratic in λ. What about Chernoff? Azuma? Chernoff assumes independence, which concerns all the moments (expectation, variance, third order moment, onwards ..). In its proof where we used the *moment generating function*, implicitly we assumed that all moments exist. The result is a much stronger exponential fall-off. Azuma's inequality is equally strong but with a weaker assumption of simply *bounded martingale*.

6. Randomized algorithms:

(a) (ADDED) general techniques:

i. boosting (for one-sided error such as Karp-Rabin string matching algorithm, see HW #7 P1; for two-sided error, such as the BPP algorithm, see HW #9 P4)

ii. alteration (in designing an indept set, see HW #4 P5)

(b) Karger's mincut algorithm:

i. recall the notion of edge contraction

ii. recall that a multi-edge may result from contraction. Is the probability of contraction for the endpoints of a multi-edge higher than those of a single edge? (... hint: the answer is yes, because every edge in the multi-edge gets a share in the probability of being picked for contraction)

iii. what is the survival probability of a mincut throughout the contraction process? (... hint: the answer is $1/\binom{n}{2}$, the derivation is in LN #4, page 4)

iv. how many min cuts can there be in a graph? (see HW#4, P1)

(c) MST (minimum spanning tree): in LN #9

i. recall the greedy property that underlies all three algorithms

ii. recall the original Boruvka's algorithm and its time complexity

iii. why can the randomized algorithm perform better than the original algorithm? where did the randomization occur in the algorithm? (... hint: if we can cut down the number of edges (as well as the number of vertices) after each recursion, we are in a better shape for a reduced time complexity. the detour through random sampling of subgraph and MST verification serves to prune the edges that won't appear in the MST)

(d) (ADDED) Byzantine agreement:

i. recall the purpose of the algorithm is for all the *good* processors to agree on a common number. (and it is not important what that number is)

ii. in the algorithm on page 360 of M&R, how large can be the fraction of faulty processors?

iii. recall the coin flip is "global" and the outcome is transmitted securely to all the "good" processors.

iv. why must "higher threshold" H be greater than the "lower threshold" L by an amount of t, the number of faulty processors?

(e) Probabilistic proofs:

i. how did we used probabilistic argument for proof of existence? recall the MAX-SAT problem (hint: .. in LN #5) recall the optimal coding theorem (... where we stated the properties of a "typical set" and proved its properties suitable for optimal coding. see LN #8)

ii. derandomization: from proof to construction how did we use the method of conditional expectation to find a Boolean assignment that satisfies a certain fraction of clauses in MAX-SAT? what property of conditional expectation enables us to proceed with this type of derandomization? (... hint: see LN #5 for derandomization of MAX-SAT by walking down a binary tree; the crucial property is the double expectation theorem, which shows that the expectation at the parent node is a weighted average of the conditional expectations at children's)

(f) the phenomenon of thresholding in random graphs:

i. recall the $G(n,p)$ model. how large does p has to be for there to exist a clique of size k? (hint:... see LN #7)

ii. furthermore, what is the size of the largest clique? more precisely, as a random variable, how does the size of the largest clique behave? (hint:... like a step function, review page 4 of LN #7)

iii. recall your attempt in finding the largest clique (HW #6 P3). why are you asked to come up with your own heuristics? (hint: ... because even the current best algorithm can only get a clique of size $\lg n$, which is a half of the threshold we got in LN#7)

(g) (ADDED) Random Walks on Graphs

i. Imagine yourself a particle walking on a graph, what property should the graph have if you want to visit all vertices, what property should the graph have if you don't want to oscillate between two parts of the graph every other step?

ii. Recall that the random walk on a directed graph is equivalent to a Markov chain (LN #11, and RN #9) Think about how you would convert one problem instance to the other.

iii. recall and compare the following three notions: hitting time, commute time, and cover time (two kinds of cover time).. is hitting time symmetric?

iv. recall the linkage between electric networks and commute times. Name a few examples where the analog of resistor network helped us compute the commute/hitting time very easily. (chain, cycle, lollipop, etc.)

v. what's an upper bound on the cover time? how was the minimum spanning tree used in the proof? (see LN #11, also HW #8 P2)

vi. we know 3SAT is NP-complete. What about 2SAT? Name a randomized algorithm that solves 2SAT efficiently. Use results from a random walk on a chain.

(h) (ADDED) fingerprinting (LN #10):

i. In general, if two objects are different, will their fingerprints always be different? Conversely, if two objects are the same, will their fingerprints always be the same? (hint: .. one-sided error)

ii. name two types of finger prints. (hint: for example, evaluate a polynomial at a set of points, the values of the polynomial is a fingerprint, what happens when the number of points exceeds the degree of the polynomial? ... recall Lagrange's interpolation, which also appeared late in the course in Shamir's method for secret sharing. Another example of fingerprinting is modulo n, recall string matching)

iii. recall the algorithm for checking polynomial identity. What's the error probability? How did we bound it? (recall: the theorem of Schwarz and Zippel on a *multivariate* polynomial. the inductive proof is in LN #10)

iv. recall the Karp-Rabin string matching algorithm. How did we bound the error probability there? What is chosen at random and what are given as constants? Why did the prime number theorem come in? (you don't have to memorize the details of the proofs, but you should make it clear what are *random* variables in the proof. Once you have a conceptual grasp of the algorithm, and you may invoke it as a black box for other problems without worry)

7. (ADDED) Number Theory & Cryptography

(a) Number Theory

i. powering: how would you compute

$$a^{1024} \mod n$$

where a and n are arbitrary integers? (hint: ... repeated squaring. this makes "powering" a polynomial-time operation, polynomial in $\log e$, where e is the exponent.) Now, if n is a prime, what property can you exploit to ease the computation? (hint: ... Fermat's little theorem) Further, if n is a composite number but you know the value of totient function $\phi(n)$, how can you ease the computation? (hint:... Euler's theorem is a generalization of Fermat's little theorem)

ii. Euler's totient function: if you know the factoring of an integer, you will know the value of its *Euler's totient function*

$$\phi(n) = n(1-\frac{1}{p_1})...(1-\frac{1}{p_k})$$

where $p_1,...,p_k$ are distinct prime factors of n.

iii. Where did we use the totient function in RSA protocol? How can you use the preceding claim about totient function to break the RSA protocol?

iv. Why is RSA protocol considered safe? (hint: ... people think integer factoring is hard) Recall that the two following questions are still open: Is integer factoring necessary for breaking the RSA protocol? Is integer factoring NP-Complete? (we don't know the answers to both)

v. discrete log: recall that discrete log is the inverse operation of powering. analogously to RSA, we don't know if discrete log is necessary for breaking El Gamal, and we don't know if discrete log is NP-Complete.

vi. reciprocal: how would you compute

$$a^{-1} \mod n$$

even more basic, when does the reciprocal exist? (hint: .. when a and n are coprime) Recall Bezout's theorem and Euclid's algorithm for computing the reciprocal. (HW #9 P5) Just like powering, this is again an "efficient" procedure that runs in polynomial time.

(b) Cryptography: (based on the constructs from Number Theory)

i. Check the steps of the RSA protocol and reason why each step takes only polynomial time.. (hint: powering and reciprocal are polynomial-time operations)

ii. Do the same for El Gamal

iii. Why is RSA vulnerable for short messages? (.. hint: consider the cracking technique of "exhaustive search". say the message only consists of two alphabetical letters. An attacker can enumerate all 26×26 possibilities, encrypt using receiver's public key, and check if the outcome matches the encrypted message on the wire) How is this remedied by El Gamal's protocol?

iv. How can you use RSA for digital signature? (.. hint: the encrypted outcome can be thought of as a "checksum" of the message)

v. How many points are needed to determine a polynomial of degree m? How did we make use of the classic Lagrange's interpolant in secret sharing? (cf. Schwarz-Zippel algorithm for polynomial identity)

vi. Recall that threshold decryption is a blend of RSA(or El Gamal) and secret sharing of the private key. Note that in the process of decryption, the secret, i.e. the private key, is not compromised and can be re-used.

vii. ZKP (zero-knowledge proofs). Recall the classical example of 3-coloring: the prover needs to convince the verifier that s/he knows that coloring without revealing it. What's the probability that the verifier is fooled? (... hint: recall that the prover is required to "commit". As an analog, consider that the prover sends to the verifier locked boxes that contain the coloring for each node, and then the verifier asks for the keys to some of the boxes. the prover has "committed" the coloring once s/he sends over the locked boxes)

viii. Recall the various places in digital cash where we used the building blocks of public-key encryption, digital signatures, secret sharing and ZKP.