1 In a graph G with n vertices and t distinct triangles (i.e., a triangle is a cycle of length 3) not necessarily disjoint, we have an induced subgraph with no triangles and with (bigger is better [if correct, naturally]):

1.A. $\geq n/t^{1/3} - 1$ vertices

1.B. $\geq n/\sqrt{t} - 1$ vertices

1.C. $\geq n/t - 1$ vertices

1.D. $\geq n - 1$ vertices

2 Given an array $A[1 \ldots n]$ with n distinct real numbers, one can output the kth smallest element in A using (in expectation [more precise bounds, if correct, are better]):

2.A. $2n + o(n)$ comparisons.

2.B. $(3/2)n + o(n)$ comparisons.

2.C. $O(\log n)$ comparisons.

2.D. $O(n)$ comparisons.

3 Considering throwing n balls into M bits, uniformly, randomly and independently. What is the expected number of bins that have exactly two balls?

3.A. $\binom{n}{2}(1/M)^2(1 - 1/M)^{n-2}$.

3.B. $\binom{M}{2}(1/n)(1 - 1/n)^{n-2}$.

3.C. $\binom{n}{2}(1/M)(1 - 1/M)^{n-2}$.

3.D. $M(1 - 1/M)^{n-2}$

3.E. $M(n/M)^2 \exp(-n/M)/2$.

4 You are given a set of n elements $e_1, \ldots, e_n$. Consider the following randomized algorithm – in step one, it creates a node n_1 that stores e_1. In the jth step, it creates a node n_j (that stores e_j), randomly picks two random numbers $\alpha_j, \alpha'_j \in [\![i-1]\!] = \{1, 2, \ldots, i-1\}$, and creates a directed edge from n_j to $n_{\min(\alpha_j, \alpha'_j)}$.

This algorithm computes a random directed tree T with n nodes (the tree is a *reverse tree*). Let L be the length of the longest path in T. The length of L is

4.A. $\Theta(\log n)$ with high probability.

4.B. $\Theta(\log \log n)$ with high probability.

4.C. $\Theta\left(\frac{\log n}{\log \log n}\right)$ with high probability.

4.D. $\Theta(\log^2 n)$ with high probability.

4.E. All of the above.

5 Let $X_1, X_2, \ldots, X_n$ be n independent random variables, such that $\mathbb{P}[X_i = 1] = 1/\sqrt{n}$, and $\mathbb{P}[X_i = 0] = 1 - 1/\sqrt{n}$. Then, for the random variable $Y = \sum_i X_i$, we have that

5.A. $\mathbb{P}[|Y - \sqrt{n}| > \log n] \leq 1/n^4$.

5.B. $\mathbb{P}\left[|Y - \sqrt{n}| > n^{1/4}\right] \leq 1/n^4$.

5.C. $\mathbb{P}\left[|Y - \sqrt{n}| > 30n^{3/8}\log n\right] \leq 1/n^4$.

5.D. $\mathbb{P}\left[|Y - \sqrt{n}| > 30n^{1/8}\log n\right] \leq 1/n^4$.

6 Considering flipping an unbiased coin enough times till one gets n heads. Let M be the number of flips. We have that

6.A. $\mathbb{P}[M > 2n] < \exp(-n)$.

6.B. $\mathbb{P}[M > 16n] < \exp(-n)$.

6.C. $\mathbb{P}[M > n] < \exp(-n)$.

6.D. $\mathbb{P}[M > n] < 1/2$.

7 Imagine throwing n balls into n bins, where you try $d = \lceil 20\log_e n \rceil$ different locations, and place the ball in the bin with lowest load. With high probability, this algorithm has maximum load of

7.A. 2.

7.B. 3.

7.C. 10.

7.D. $O(\log\log n)$.

7.E. $O(\log n/\log\log n)$.

7.F. who knows or cares? Not me!

8 Imagine throwing $\lceil 20n\log_e n \rceil$ balls into n bins with 2-choices. Then with high probability, we have

8.A. The maximum load of a bin is $O(\log n)$.

8.B. The maximum load of a bin is $O(\log n/\log\log n)$.

8.C. The maximum load of a bin is $O(\log n \log\log n)$.

8.D. The maximum load of a bin is $O(\log^2 n/\log\log n)$.

8.E. The maximum load of a bin is $O(\log^2 n)$.

9 Imagine throwing n balls into n bins. If the chosen bin for a ball has more than two balls, the algorithm tries again, until it finds a bin with less then two balls, where it places the ball. This algorithm would (in expectation) overall try

9.A. $\Theta(n\log n)$ bits.

9.B. $\leq 2n$ bits.

9.C. $\sum_{i=1}^{n}\frac{2}{i}$ bits.

9.D. $O(n^2)$ bins.

9.E. Nobody knows, its a mystery wrapped in an enigma, buried undergrad, in an unmarked cave.

10 Let S be a stream of m numbers taken from $\{1,\ldots,n\}$. How much space is needed to provide an $(1 \pm \varepsilon)$-estimate that is correct with high probability for F_3 of the stream?

10.A. $O(\varepsilon^{-2} n^{2/3} \log n)$.

10.B. $O(\varepsilon^{-2} n^{1/3} \log n)$.

10.C. $O(\varepsilon^{-2} \sqrt{n} \log n)$.

10.D. $O(\varepsilon^{-2} \log n)$.

10.E. $O(\varepsilon^{-2} \log n \log \log n)$.

11 GUESTIMATE THIS! Let S be a set of n elements, and $U \subseteq S$. Assume that given an element $s \in S$, you can check in constant time, if $s \in U$. Let $p \in (0,1)$ be a parameter, and assume that $|U| \geq pn$. Then, computing a number β, such that

$$(1-\varepsilon)|U| \leq \beta \leq (1+\varepsilon)|U|,$$

can be done, by sampling τ elements from S, checking how many of them are in S (denote this number by X), and then outputting the number $\beta = n(X/\tau)$. What τ has to be for the result to be correct with high probability?

11.A. $O(\frac{1}{\sqrt{p\varepsilon}} \log n)$.

11.B. $O(\frac{1}{\varepsilon^2} \log n)$.

11.C. $O(\frac{1}{p^2\varepsilon^2} \log n)$.

11.D. $O(\frac{1}{p\varepsilon^2} \log n)$.

11.E. $O(\frac{1}{p\varepsilon} \log n)$.

12 You are given a stream S of m numbers, and a parameter ε. How much space does one need to output a number x in the stream, such that the rank of x is in the range $(1-\varepsilon)n/2 \ldots (1+\varepsilon)n/2$ with high probability? (As usual, smaller is better, if correct.)

12.A. $O(1)$

12.B. $O(1/\varepsilon)$

12.C. $O((1/\varepsilon^2) \log n)$

12.D. $O((\sqrt{n}/\varepsilon^2) \log n)$

12.E. $O((1/\varepsilon^{3/2}) \log n)$

13 Consider the task of computing a satisfiable assignment for a given 3-SAT F (which we assume is satisfiable), where F has n variables and m clauses. This can be done by a randomized algorithm (with high probability) in

13.A. $(n+m)2^{O(\sqrt{n})}$ time.

13.B. $(n+m)2^{O(\sqrt{n})}$ time.

13.C. $(n+m)2^{O(n^2)}$ time.

13.D. $O((n+m)^3 2^{cn})$ time, where $c < 1$ is some constant.

13.E. Nobody knows.

14 Consider an urn with n balls, where $n/10$ of them are black, and the rest are snow white. In each iteration of the game you pick two balls. If exactly one of them is black, then you replace it by a new white ball. Otherwise, you continue to the next iteration. How many iteration does one has to do this process till all the ball are white.

14.A. $O(n^{1.1} \log n)$

14.B. $O(n)$

14.C. $O(n \log n)$

14.D. $O(n^2)$

14.E. $O(n\sqrt{\log n})$

15 You are given $t = O(n^3)$ subsets $S_1, S_2, \ldots, S_t \subseteq [\![n]\!]$, such that $|S_i| \leq n/2$, for all i. Consider the task of building a graph G over $[\![n]\!]$, such that for all i, we have that $G - S_i$ is still connected (i.e., the graph resulting from deleting from it all the vertices of S_i). What is the number of edges of G needed to have this property?

15.A. $O(n^2)$

15.B. $O(n \log n)$

15.C. $O(n \log^2 n)$

15.D. $O(n \log^3 n)$

15.E. $O(n^3)$

16 The JL-lemma preserves the area of triangles with high probability. That is, consider a set P of n points in $\mathbb{R}^d$, and let M be a random-projection matrix as generated by the JL-lemma. Then, for any three points $p, q, r \in P$, we have

$$\text{area}(\triangle pqr) \in (1 \pm c\varepsilon)\text{area}(\triangle p'q'r').$$

Where $p' = Mp$, $q' = Mq$ and $r' = Mr$, and c is some absolute constant. This claim is

16.A. True.

16.B. False.

16.C. Undecidable.

17 Let $S_1, \ldots, S_u \subseteq [\![n]\!]$ be subsets of $\{1, \ldots, n\}$ each of size $n/2$. Consider a random sample $R \subseteq [\![n]\!]$ of size t where each element is picked uniformly and indecently with repetition from $[\![n]\!]$. The expected number of sets S_i such that $S_i \cap R = \emptyset$ is exactly

17.A. u/t.

17.B. $u - u/t^2$.

17.C. u/t^2.

17.D. $u - u/t$.

17.E. $u/2^t$.

18 You have n patients $[\![n]\!]$, and k of them are sick (you do not know the value of k). You can do group testing, and check for any specified subset $G \subseteq [\![n]\!]$, if there any sick person in G. This requires one group tests. Consider any algorithm that outputs a subset of $[\![n]\!]$ of size $n/2$ of people that are healthy. In expectation, how many group tests does the algorithm has to perform? (Smallest is better if correct.)

18.A. $O(k)$

18.B. $O(\log n)$

18.C. $O(k \log n)$

18.D. $O(\log(nk))$

18.E. $\pi^2/6$.