COL866: Special Topics in Algorithms
Concentration inequalities and their Applications in Computer Science
II semester, 2022-23.
Minor 3 exam.
30 March 2023, Maximum Marks: 25.

Instructions: Please handwrite your solutions clearly and concisely. Try to fit the solution for each problem in one side on an A4 sized page, definitely no more than 2. Latexed solutions are highly encouraged. Upload the paper on gradescope by the deadline (no extensions will be given under any circumstances). Make sure you mark the pages related to each problem on gradescope.

Problem 1 (6 marks)
Consider the class $\mathcal{F}$ of functions $f : \mathbb{R}^n \to \mathbb{R}$ that are Lipschitz w.r.t. the ℓ_1 norm, i.e., given $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, \ldots, y_n)$, $|f(\boldsymbol{x}) - f(\boldsymbol{y})| \leq \sum_{i=1}^{n} |x_i - y_i|$. Let $X = (X_1, \ldots, X_n)$ be a vector of independent random variables with finite variance. Use the Efron-Stein inequality to show that the maximal value of $\mathrm{Var}\,(f(X))$ over $f \in \mathcal{F}$ is attained by the function $f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i$.

Problem 2 (6 marks)
Recall that the E-R random graph $G_{n,p}$ is formed by taking n vertices and including each undirected edge with probability p independent of all other edges. We say that a vertices u, v, w form a triangle in a graph if they form a 3-clique in the graph. Directly calculate the variance of the number of triangles in $G_{n,p}$. Compare the variance to the upper bound obtained by using the Efron-Stein inequality.

Problem 3 (5 marks)
Suppose we are given n bins. We throw t red and t blue balls into the bins independently and uniformly at random. Using the method of bounded differences, find a value of t such that with probability at least $1 - 1/n$ there is a bin that contains both a red and a blue ball.

Problem 4 (8 marks)
Given a graph $G = (V, E)$ and some $k > 0$ we say that a function $f : E \to [k]$ is a valid edge colouring of G if all edges incident on any vertex have different colours. We will now describe a randomised algorithm to colour a bipartite graph $G = (V, E)$ with equal sized partitions V_1 and V_2 such that $|V_1| = |V_2| = n$. Further we will assume each vertex has the same degree Δ. For a fixed $C \geq 0$ the algorithm is as follows:

Require: Given a bipartite graph $G = ((V_1, V_2), E)$ and an integer $C > 0$.
Set $i \leftarrow 0$; $E_i \leftarrow E$
while $|E_i| > C$ **do**
  Choose a tentative colour i.i.d. from $\{i\Delta, i\Delta + 1, \ldots, (i+1)\Delta - 1\}$ for each edge in E_i.
  For each vertex $v \in V_1$ check if there are multiple edges of the same tentative colour. If so, choose one of them at random, retain its tentative colour and uncolour all the others.
  For each vertex $v \in V_2$ check among the edges that are still (tentatively) coloured if there are multiple edges of the same tentative colour. If so, choose one at random, fix its colour and uncolour all the others.
  Add to E_{i+1} all the edges of E_i that are still uncoloured.
  Set $i \leftarrow i + 1$.
end while
Colour the remaining edges with at most C previously unused colours.

How many colours does this algorithm require to edge colour the graph? Your answer should be a high probability answer, i.e., the bound should hold with probability at least $1 - o(1)$. Clearly the bounded differences inequality will be used here. You will have to figure out what is the best value to use for the constant C used by the algorithm. Note this is a slightly open-ended problem where you may have to spend some time to figure out how to proceed. So plan accordingly.