# HW 1

Due on **Monday, February 12, 2024 at 10am**

**CS 574: Randomized Algorithms, Spring 2022** Version: **1.11**

The homeworks can be submitted in groups of any size, as long as the size is an integer of value at most 3. The solution should be typed using latex (with no errors), and the pdf should be submitted electronically on gradescope.

If you get a considerable amount of information from somebody you should mention it in your solution (which is quite alright). As for google policy, a lot of homework questions might have solutions on the web (hopefully not, but I can not be sure). I would *prefer* if you try to solve them without searching on the web, but if you use solutions found on the web, you should **explicitly** say so in your solution (and provide the link!). In any case, I expect you to write your solution on your own (i.e., cut and paste is not acceptable), since the homeworks are there to demonstrate that you know the material and understand it.

If you have any questions, feel free to email me.

**Grade policy.** Homeworks would be 25% of the final grade. The main purpose of the homeworks is to prepare you for the final. I would recommend that you individually spend significant time trying to solve the homeworks on your own, but this is up to you.

**Solutions.** No solutions to the homeworks would be posted. If you want to know a solution to a question, ask me in person.

**Planned # of homeworks would be in the range 7-10.**

(Your solutions should be elementary and self contained. Do not use Chernoff inequality or Martingales in solving these problems, or any tool not seen in class.)

## 1 (100 PTS.) Independent.

Let $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ be a graph over $n$ vertices and $m$ edges.

**1.A.** (30 PTS.) Consider the experiment, where you pick every vertex of $\mathsf{V}$ into $U$ with probability $p$. What is the expected number of vertices/edges in the induced subgraph $\mathsf{G}_U = (U, \{uv \in \mathsf{E} \mid u, v \in U\})$?

**1.B.** (30 PTS.) Consider the algorithm for computing an independent set in $\mathsf{G}$, that first computes $U$ as described above, and then delete one of the endpoints of every edge of $\mathsf{G}_U$. In the end of this process, the surviving elements of $U$ are independent set in $G$. What choice of $p$ maximizes the expected size of the independent set being computed? (Let $I$ Be the size of this independent set.) What is $\mathbf{E}[I]$? Provide exact bounds if possible, or upper/lower bounds as tight as possible.

**1.C.** (40 PTS.) Assume the graph $G$ has $t$ triangles (i.e., cycles of length 3). Describe an algorithm that follows the above scheme (i.e., pick a random sample of vertices, delete a vertex from any induced triangle), and outputs an induced subgraph, with as many vertices as possible, that contains no triangle. What is the expected number of vertices in the graph output by your algorithm as a function of $n$ and $t$? What is the expected running time of your algorithm? (Faster is better.)

## 2 (100 PTS.) Quick select, but better.

It is not hard to prove that **QuickSelect** performs in expectation $O(n)$ comparisons (see for example the class notes). Consider the modified version of **QSelectBetter**$(A[1 \ldots n], t)$. The task at hand is to output the $t$th smallest element in $A$ (assume all the elements are distinct). Let $k$ be a fixed constant. The algorithm **QSelectBetter** picks randomly $k$ elements from $A$, compute their median (by, for example, sorting the $k$ elements and returning the middle one), and uses this element as a pivot, recursing in the natural way on the subproblem containing the $t$th smallest element. Using Chebychev inequality, **prove** that for any $\delta \in (0, 1)$, there is a value of $k$ (which is a function only of $\delta$), such that the expected number of comparisons performed by the **QSelectBetter** algorithm is at most $(2 + \delta)n + o(n)$. What is the bound on $k$ as a function of $\delta$?

## 3 (100 PTS.) Bins, balls, birthdays.

**3.A.** (20 PTS.) You throw $t$ distinct balls into $n$ distinct bins (each ball picks uniformly at random the bin its being thrown into). Provide tighter upper/lower bounds on the value of $t$, such that with probability at least $1/2$, there are some three (or more) balls that lie in the same bin.

(One direction is easy, the other one requires some more work.)

**3.B.** (80 PTS.) Imagine playing the above game, but discarding the balls that are in bins that contain three or more balls. Thus, you start with $n$ balls and each round some balls get discarded. Provide an upper/lower bounds on the expected number of rounds one has to play till at most 2 balls remain.