HW 2

Due on **Monday, February 28, 2022 at 10am**

CS 574: Randomized Algorithms, Spring 2022 Version: **1.2**

Submission guidelines same as previous homework.

4 (100 PTS.) Ball throwing mayhem.

4.A. (50 PTS.) You are given n balls $b_1, b_2, \ldots, b_n$, where the ball b_i is labeled with the number i. There are also n bins $B_1, \ldots, B_n$, which are also labeled in the natural way by $1, \ldots, n$.

The game begins as follows: The first ball b_1 picks a random bin to store itself in. For $i > 1$, we check for the ith ball b_i if B_i is empty. If so, we put b_i in B_i. Otherwise, we randomly choose a location ℓ_i out of the empty bins, and put b_i in the bin B_{ℓ_i}.

Consider the last ball b_n. What is the probability that b_n is stored in the bin B_n?

[This is a tricky question – there is a short and elegant solution, but it is not easy to find.]

4.B. (30 PTS.) Let us repeat the above game, but the first k balls randomly choose locations, and the later balls try to go to their designated bin, and if their designated bin is non-empty, then they pick an empty bin at random. What is the probability of b_n to be stored in the bin B_n?

4.C. (20 PTS.) In the settings of (4.B.), what is the probability that, for all j, the ball b_j is stored in the bin B_j, for $j = n - k + 1, \ldots, n$?

5 (100 PTS.) Smallest k distances.

You are given a set P of n points in the plane. Let D be the set of $\binom{n}{2}$ pairwise distances between any pair of points of P (assume all these $\binom{n}{2}$ values are distinct). Given a parameter k (think about k as being relatively small), describe an algorithm, as fast as possible (in expectation, say) that outputs the kth smallest number in D.

(Running time of $O(n + k^2)$ in expectation is doable, but is probably too hard. An algorithm with $O(nk)$ expected running time is not too difficult. If you can do anything in between, that would be nice.)

Hint: Modify the closest pair algorithm seen in class (which solves the case $k = 1$).

6 (100 PTS.) More balls into bins.

6.A. (50 PTS.) Consider throwing n balls into n bins, where if a bin chosen (say at location B_i) is already occupied, you try the next $t - 1$ consecutive bins (i.e., $B_i, B_{i+1}, \ldots, B_{i+t-1}$) and place the ball in the first unoccupied bin found (here $i + t - 1$ is computed module n, so location $n + 1$ is location 1, etc). If all these bins are occupied, then the ball is rejected. Provide upper bound and lower bound (hopefully as close as possible), to the total *expected* number of balls rejected by this process, as a function of t.

6.B. (30 PTS.) We throw n pieces of chewing gum into n bins, if a gum falls into an non-empty bin, then the gum sticks to the gum already there. Let X_1 be the number of gum pieces in the end of the first round. Provide upper and lower bounds on the expectation of X_1.

6.C. (20 PTS.) Assume that you now repeat this game. In the ith iteration, you take all the gum pieces (which are the fusion of several original gum pieces) from the previous round, and throw them into n bins (which are empty again). Let X_i be the number of gum pieces in the end of this process (again, gum pieces that fall into the same bin, stick together)

A round is final, if no gum piece got bigger during this round (i.e., $X_i = X_{i-1}$).

Prove that if $X_{i-1} < \sqrt{n}$, then the ith round is final with probability larger than some constant $c > 0$ (what is the value of c in your analysis). Similarly, prove that with high probability, if $X_{i-1} > 20\sqrt{n}\ln n$, then the next round is not final.