Problem Set 1 – Due Thu, 4 Apr 2024

Reminders: If you wish to turn in anything, you do so just before discussion-section begins on Thursday. It must be correct, succinct, and typset in LaTeX, You may not employ old ECS127 problem-set solutions in any manner. Clearly state any (reasonable) assumptions you must make.

Problem 1. Alice has a pretty penny. Unfortunately, it might not be a *fair* penny: it might, when flipped, land heads with some probability $p \neq 0.5$. Alice wants to generate a uniform random bit b: the bit should be 1 with probability 0.5 and zero with probability 0.5. Describe a strategy Alice can use to achieve the result she wants using her possibly-biased coin.

Problem 2. Alice and Bob have an infinite pile of pennies. They take turns placing their pennies on a perfectly round table, beginning with Alice. A penny may be placed anywhere on the table so long as all of the penny fits fully on top of the table and no part of the penny is on top of any other penny. Pennies must be placed flat on their heads or tails side. A party loses if he has nowhere to put his penny. Show that Alice can always win. (You might need some natural assumption for this to be true. If so, state it.)

Problem 3. Alice might like to go on a date with Bob. Bob might like to go on a date with Alice. But nobody asks the other out because they're too embarrassed to express interest in case the other is not interested.

Alice and Bob aim to solve this problem by designing a protocol (an algorithm) in which each learns of the other's interest if *both* are interested. Said differently, Alice has a private bit $a \in \{0,1\}$ and Bob has a private bit $b \in \{0,1\}$, and we seek a method wherein Alice and Bob can interact with one another and, at the end of the interaction, each will know $a \wedge b$, but nothing more. If $a = b = 1$ they each learn this fact; if $a = 0$, Alice learns nothing of b; if $b = 0$, Bob learns nothing of a.

For your solution, use only simple, physical objects you might find around your home. Assume Alice and Bob are basically honest and cooperative, but don't assume either will do what you say if left unobserved.

Problem 4. An n-bit permutation P is a one-to-one and onto function with domain and range $\{0,1\}^n$. The set of all n-bit permutations is denoted $\mathrm{Perm}(n)$. By a random n-bit permutation I mean a function drawn uniformly from $\mathrm{Perm}(n)$.

An n-bit cycle C is an n-bit permutation for which $0^n, C(0^n), C(C(0^n)), \ldots, C^{2^n-1}(0^n)$ are distinct. The set of n-bit cycles is denoted $\mathrm{Cycl}(n)$. By a random n-bit cycle I mean function drawn uniformly from $\mathrm{Cycl}(n)$.

(As a suggested warm-up, draw some pictures illustrative of random permutations and random cycles; figure out why, for a random cycle, $C^{2^n}(0^n) = 0^n$; and compute $|\mathrm{Perm}(n)|$ and $|\mathrm{Cycl}(n)|$.)

Finally, the question: Fix $n \geq 1$. Now show how to convert a random permutation $P \in \mathrm{Perm}(n)$ into a random cycle $C \in \mathrm{Cycl}(n)$. That is, provide a (stateless, deterministic) algorithm to compute $C(x)$ that makes (efficient, black-box) use of permutations $P(y)$ and $P^{-1}(z)$. Explain why C is a cycle, and why it is uniformly random in $\mathrm{Cycl}(n)$ as long as P is uniformly random in $\mathrm{Perm}(n)$.