Warm-up

Problem 1. Check your understanding: how many independent random bits are necessary (and sufficient) to generate a uniformly random integers in $\{1, \ldots, n\}$? To generate a uniformly random subset $S \subseteq \{1, \ldots, n\}$?

Problem 2. Let X, Y be independent Bernoulli random variables with parameter $1/2$ (that is, independent, uniformly random bits), and set $Z = X \oplus Y$. Show that Z is a uniformly random bit, and that X, Y, Z are pairwise independent but not independent.

Problem 3. We have seen in class the definition of a family of pairwise independent hash functions, also called a *strongly universal hash family* from $\mathcal{X}$ to $\mathcal{Y}$: $\mathcal{H}$ is such a family if, for every distinct $x, x' \in \mathcal{X}$ and every $y, y' \in \mathcal{Y}$, we have

$$\Pr_{h \sim \mathcal{H}}[\, h(x) = y, h(x') = y' \,] = \frac{1}{|\mathcal{Y}|^2}$$

where the probability is over the uniformly random choice of $h \in \mathcal{H}$. We now introduce a related (but weaker) concept: $\mathcal{H}$ is a *universal hash family* from $\mathcal{X}$ to $\mathcal{Y}$ if, for every distinct $x, x' \in \mathcal{X}$,

$$\Pr_{h \sim \mathcal{H}}[\, h(x) = h(x') \,] \leq \frac{1}{|\mathcal{Y}|}$$

Show that every strongly universal hash family is a universal hash family. *(Note: the converse is not true, see for instance Problem 8.)*

Problem solving

Problem 4. Give a randomised algorithm which, on input a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, runs in time $O(m(n+m))$ and outputs a cut (A, B) such that $c(A, B) \geq \frac{m}{2}$ with probability at least 0.99.

Problem 5. We will prove Fact 22.2 from the lecture notes:

> There exists an explicit family of pairwise independent hash functions $\mathcal{H} \subseteq \{h \colon [n] \to \{0,1\}\}$ with $|\mathcal{H}| = 2^{\lceil \log(n+1) \rceil}$.

To do so, suppose for simplicity that $n+1$ is a power of 2, i.e., $n = 2^k - 1$ for some integer k. We will identify an integer $1 \leq x \leq n$ with its binary representation $x \in \{0,1\}^k$ (note that this representation is *not* the all-zero vector, as $x \neq 0$). Define $\mathcal{H} = \{h_S\}_{S \subseteq \{0,1\}^k}$, where, for a given set $S \subseteq \{0,1\}^k$,

$$h_S(x) = \bigoplus_{i \in S} x_i, \qquad x \in \{0,1\}^k$$

that is, $h_S(x)$ is the sum, modulo 2, of the bits of x that are indexed by S.

a) What is the size $|\mathcal{H}|$ of $\mathcal{H}$?

b) How many random bits does it take to draw a hash function h from $\mathcal{H}$? Argue such a has function can be drawn, stored, and evaluated (on any input x) efficiently.

c) Show that $\mathcal{H}$ is a family of pairwise independent hash functions.

Problem 6. In an (undirected) graph $G = (V, E)$, a *triangle* is a triple of vertices u, v, w such that the 3 edges $(u, v), (v, w), (u, w)$ exist in E. In a *directed* graph $G = (V, \vec{E})$, an oriented triangle is a cycle of length 3: namely, a triple of vertices u, v, w such that the 3 directed edges $(u \to v), (v \to w), (w \to u)$ exist in $\vec{E}$.

Given as input an undirected graph G, we want to give an orientation to each edge $e \in E$ (that is, convert G into a *directed* graph) while maximising the number of oriented triangles in the resulting directed graph.

a) Give a randomised algorithm whose output has an *expected* number of oriented triangles at least $1/4$ the maximum possible number $\mathrm{OPT}(G)$.

b) Convert your algorithm into a *deterministic* (efficient) algorithm achieving the same approximation guarantee.

Problem 7. Given a 2-colouring $c\colon E \to \{\text{red}, \text{blue}\}$ of a graph $G = (V, E)$, a *monochromatic triangle* is a triple of vertices $(u, v, w) \in V^3$ such that the edges $(u, v), (v, w), (u, w)$ exist (are in E) and $c(u, v) = c(v, w) = c(u, w)$ (they have the same colour). Show that, for every n, there exists is a 2-colouring of the complete graph K_n with *at most* $\frac{n^3}{24}$ monochromatic triangles. Give an efficient (polynomial-time) deterministic algorithm which, on input n, finds such a 2-colouring.

Advanced

Problem 8. Fix a prime number $p \geq 2$ and an integer $n \geq 1$. For a given $a = (a_1, \dots, a_n) \in \mathbb{Z}_p^n$, define the function $h_a\colon \mathbb{Z}_p^n \to \mathbb{Z}_p$ by

$$h_a(x) = \sum_{i=1}^{n} a_i x_i \bmod p, \qquad x \in \mathbb{Z}_p^n$$

and let $\mathcal{H} = \{h_a\}_{a \in \mathbb{Z}_p^n}$.

a) How many bits does it take to fully specify a function $h \in \mathcal{H}$? And an arbitrary function $f\colon \mathbb{Z}_p^n \to \mathbb{Z}_p$?

b) Show that $\mathcal{H}$ is a universal hash family (see Problem 3); that is, for every $x, x' \in \mathbb{Z}_p^n$,

$$\Pr_{h \sim \mathcal{H}}\left[\, h(x) = h(x') \,\right] = \frac{1}{p}$$

c) Is it a strongly universal hash family?