Homework 6:

Due: October 28, 2025 at 2:30p.m.

This homework must be typed in LaTeX and submitted via Gradescope.

Please ensure that your solutions are complete, concise, and communicated clearly. Use full sentences and plan your presentation before your write. Except where indicated, consider every problem as asking for a proof.

Problem 1. A family $\mathcal{H}$ of hash functions from U to $[0,\dots,m-1]$ is 2-universal iff for any $x \neq y \in U$ and $h \sim \text{Unif}(\mathcal{H})$,

$$\Pr(h(x) = h(y)) \leq \frac{1}{m}.$$

Show that each of the following families is *not* 2-universal.

1. Let $p > m$ be prime. $H_1 = \{\, h(x) = ((ax \bmod p) \bmod m) \mid a \in \{1,\dots,p-1\} \,\}$.
2. Let $p > m$ be prime. $H_2 = \{\, h(x) = ((x + b \bmod p) \bmod m) \mid b \in \{0,\dots,p-1\} \,\}$.
3. Let q be a multiple of m. $H_3 = \{\, h(x) = ((ax + b \bmod q) \bmod m) \mid a \in A,\ b \in B \,\}$ for any nonempty finite $A, B \subset \mathbb{Z}$ (e.g. $A = \{1,\dots,q-1\}$, $B = \{0,\dots,q-1\}$).

Solution. We give, for each family, distinct $x \neq y$ such that $\Pr(h(x) = h(y)) > \frac{1}{m}$.

(1) H_1 is not 2-universal. Take $m = 3$ and $p = 5$ (as hinted) and choose $x = 1$, $y = 4$. For $a \in \{1,2,3,4\}$:

$$h(1) = (a \cdot 1 \bmod 5) \bmod 3, \qquad h(4) = (a \cdot 4 \bmod 5) \bmod 3.$$

Evaluating:

a	1	2	3	4
$h(1)$	1	2	0	1
$h(4)$	1	0	2	1

They collide for $a = 1$ and $a = 4$, so $\Pr(h(1) = h(4)) = 2/4 = 1/2 > 1/3 = 1/m$. Hence not 2-universal.

(2) H_2 is not 2-universal. Again take $m = 3$, $p = 5$. Let $x = 0$, $y = 3$. For $b \in \{0,1,2,3,4\}$:

$$h(0) = (0 + b \bmod 5) \bmod 3, \quad h(3) = (3 + b \bmod 5) \bmod 3,$$

which gives

b	0	1	2	3	4
$h(0)$	0	1	2	0	1
$h(3)$	0	1	0	1	2

Collisions occur for $b = 0, 1$: $\Pr(h(0) = h(3)) = 2/5 = 0.4 > 1/3$. Thus not 2-universal.

(3) H_3 is not 2-universal. Since q is a multiple of m, reducing mod$q$ and then mod$m$ is the same as reducing directly mod$m$. Thus for any a, b,

$$h(x) \equiv ax + b \pmod{m}.$$

Choose any distinct x, y with $x \equiv y \pmod{m}$ (possible whenever U contains two distinct elements congruent mod m). Then for *every* a, b, $ax + b \equiv ay + b \pmod{m}$, so $h(x) = h(y)$ always. Hence $\Pr(h(x) = h(y)) = 1 > 1/m$.
Therefore none of the three families is 2-universal. □

Problem 2. Given an undirected graph G, a DFS from a vertex produces a DFS tree. Use it to detect separating (articulation) vertices.

(a) Show the DFS root is a separating vertex iff the root has at least two children in the DFS tree.

(b) Show that a non-root vertex v is a separating vertex iff there exists a child w of v such that no descendant of w has a back edge to a proper ancestor of v.

Solution. We use the standard DFS notions: tree edges, back edges, discovery times disc[$\cdot$], and low[$\cdot$] where

$$\text{low}[u] = \min\big(\text{disc}[u],\ \{\text{disc}[z] \text{ for back edges } (u,\ z)\},\ \{\text{low}[x] \text{ for children } x\}\big).$$

Recall: a vertex v is a separating (articulation) vertex iff its removal increases the number of connected components.

(a) Root criterion. *(⇒)* If the root r has at least two children c_1, c_2 in the DFS tree, then any path from a vertex in the subtree of c_1 to a vertex in the subtree of c_2 must pass through r (there are no back edges to ancestors of r). Removing r disconnects those subtrees, so r is separating.
(⇐) If the root r has at most one child, then every vertex discovered by the DFS lies in that unique child subtree; G remains connected after deleting r if and only if edges connect across that subtree, but any such edges are within the subtree. Since there is only one child subtree, removing r cannot separate two disjoint DFS subtrees; thus r is not an articulation point. Formally, with one child the DFS tree is a single tree below r, and removing r leaves it connected. Therefore, r is separating iff it has ≥ 2 children.

(b) Non-root criterion. Let v be a non-root vertex.
(⇒) Suppose there is a child w of v such that no descendant of w has a back edge to a proper ancestor of v. Then all vertices in the subtree of w can only reach ancestors in the DFS tree via v; removing v disconnects that subtree from the rest of G. Hence v is separating.
Equivalently in low-language: "no descendant of w has a back edge to a proper ancestor of v" means low[w] $\geq$ disc[v]. This is the classical articulation condition.
(⇐) Conversely, suppose v is separating. Then, in the DFS tree, there exists a child subtree T_w of v that becomes disconnected from the rest of the graph upon removing v. If some descendant of w had a back edge to a proper ancestor of v, then vertices in T_w would still connect to the rest of G avoiding v via that back edge. Hence, for such a separating child w, no descendant has a back edge to a proper ancestor of v, i.e. low[w] $\geq$ disc[v].
Thus the stated condition is necessary and sufficient. □

Problem 3. Let $G = (A, B, E)$ be bipartite with $E \subseteq A \times B$. Let $S \subseteq A$ and $T \subseteq B$. Assume there is a matching covering S and a (possibly different) matching covering T. Prove there is a matching covering $S \cup T$.

Solution. Let M_A be a matching covering S, and M_B a matching covering T. Consider the symmetric difference

$$H = M_A \triangle M_B$$

which decomposes into vertex-disjoint alternating paths and even cycles whose edges alternate between M_A and M_B.
Within each connected component C of H:

- If C is an even cycle, both $M_A \cap C$ and $M_B \cap C$ are perfect matchings of C. Either choice covers exactly the same vertex set in C (all of it).
- If C is an alternating path, its endpoints are unmatched in exactly one of M_A or M_B. There are two matchings on C: the M_A-edges or the M_B-edges along the path. Exactly one of these two matchings covers the endpoint in A and the endpoint in B that are matched in the other global matching.

Now build a new matching M componentwise as follows:

- On each cycle component, pick either side (say, $M_A \cap C$).
- On each path component C, if C contains a vertex of $S \cup T$ that is not covered by M_A (respectively M_B), choose the M_B-edges (respectively the M_A-edges) along C so that this endpoint becomes covered in M.

Because components are disjoint, this yields a valid matching M. By construction:

- Every $s \in S$ is covered: if s was already covered in M_A and lies in a cycle, it stays covered; if s lies on a path where the M_A-edge covering s would be dropped, that path necessarily has its other endpoint uncovered in M_B, so we instead select the M_A-edges on that path to keep s covered.
- Every $t \in T$ is covered by a symmetric argument.

Thus M covers $S \cup T$. □