1. (15 points) (The power of Decision Diffie-Hellman.)

In class we saw that the DDH assumption can be used for public-key encryption; here you will show that it is very useful for symmetric primitives too.

For a cyclic group $G = \langle g \rangle$ of prime order $q$, the DDH assumption says that

$$(g, g^a, g^b, g^{ab}) \approx (g, g^a, g^b, g^c),$$

where $a, b, c \leftarrow \mathbb{Z}_q$ are uniformly random and independent. By grouping the elements appropriately, we can view this assumption in matrix form:

$$g \begin{pmatrix} 1 & a \\ 1 & b \end{pmatrix} \approx g \begin{pmatrix} 1 & a \\ b & c \end{pmatrix},$$

where $g^M$ (for a matrix $M$ over $\mathbb{Z}_q$) is the matrix over $G$ obtained by raising $g$ to each entry of $M$.

Observe that in the left-hand matrix, the two rows are linearly dependent (over $\mathbb{Z}_q$), while in the right-hand matrix they are very likely not to be.

(a) (7 points) Prove that the DDH assumption implies that, for any positive integer $w = \text{poly}(n)$,

$$g \begin{pmatrix} a_1 & a_2 & \cdots & a_w \\ a_1 \cdot b & a_2 \cdot b & \cdots & a_w \cdot b \end{pmatrix} \approx g \begin{pmatrix} a_1 & a_2 & \cdots & a_w \\ c_1 & c_2 & \cdots & c_w \end{pmatrix},$$

where $a_i, b, c_i \leftarrow \mathbb{Z}_q$ are all uniformly random and independent.

(b) (4 points) Using the previous part, prove that the DDH assumption implies that, for any positive integers $w, h = \text{poly}(n)$,

$$g \left( \prod_{i=1}^{w} a_i \cdot b_j \right) \approx g \left( \prod_{i=1}^{w} c_{i,j} \right),$$

where $a_i, b_j, c_{i,j} \leftarrow \mathbb{Z}_q$ are all uniformly random and independent. Note that the left-hand matrix (in the exponent) has rank 1, while the right-hand matrix is very likely to be full-rank.

(c) (4 points) Conclude that under the DDH assumption, there is a PRG family expanding about $2n \lg q$ bits to about $n^2 \lg q$ bits. (The output need not literally be made up of bits, though.) For the same input and output lengths, why might we prefer this PRG to the one from class based on a OWF?

(d) (0 points) (Challenge question.) Generalize the above to design a pseudorandom function based on DDH. (Hint: extend to $2 \times 2 \times \cdots \times 2$ matrices.)

2. (4 points) (Authentication.\footnote{From Peikert}) A message authentication code (MAC) allows two parties sharing a secret key to check that messages they exchange have not been tampered with during transmission. Show that one-time pad (OTP), despite offering perfect security, is not enough for this task.

Next, let us try to define MACs formally. The model is this. We have an algorithm $Gen$ that outputs a random key $k$, an algorithm $Tag$ that takes a key $k$ and message $m$ and outputs a “tag” $t$, and an algorithm $Ver$ that takes a key $k$, a message $m$, and a tag $t$, and either accepts or rejects. The correctness requirement says that for any message $m$, key $k$, if $t = Tag_k(m)$ then $Ver_k(m, t)$ accepts. Suggest some notions of security and compare them. Some things to consider: information theoretical vs. computational; what constitutes a valid forgery? what does the attacker get to see or allowed to do?