Cryptography--ENEE/CMSC/MATH 456
Class Exercise--2/13/19

Let G be a pseudorandom generator where $|G(s)| = |s| + 1$

1. Define $G'(s) = G(s||\overline{s})$, where $\overline{s}$ is the bit-wise negation of s. Is G' necessarily a pseudorandom generator?

   Short answer: No. $s||\overline{s}$ is not uniformly distributed and the guarantees of a PRG hold only when its input is uniformly distributed. To see why it is not uniformly distributed: Assume s has length n. The number of elements in the set of strings of the form $s||\overline{s}$ is only 2^n, whereas the number of $2n$-bit strings is 2^{2n}. Therefore, we are sampling from a $1/2^n$-fraction of the total number of strings.

2. Define $G'(s) = G(s)||G(\overline{s})$, where $\overline{s}$ is the bit-wise negation of s. Is G' necessarily a pseudorandom generator?

   Short answer: No. We can generate multiple pseudorandom strings using a PRG, but each time we run the PRG the seed must be chosen uniformly at random and independently from all other seeds. In this case, each of $s, \overline{s}$ are individually uniform random, but they are not independent.

3. Define $G'(s) = G(s)_1||G(G(s)_2, \ldots, G(s)_{|s|+1})$, where $G(s)_i$ denotes the i-th output bit of $G(s)$. Is G' necessarily a pseudorandom generator?

   Short answer: Yes. We are basically running a PRG on the output of a PRG. We can argue as follows: The output of the first invocation of the PRG is pseudorandom since the seed s was uniform random. The second invocation does not get a uniform random input, but a pseudorandom input. But pseudorandom is as good as random when we are concerned with poly-time distinguishers only (as is the case here). So the output of the second invocation is also pseudorandom.