**Problem 1:** Your friend shows you an algorithm INV that can invert a given function f in PPT, but only for those f(x) where the first 20 high-order bits of x are all 1's. However, if the first 20 bits are not all 1's, you assume that for any PPT adversary A, the probability that A inverts F becomes negligible as the length of the input increases. Could f be a one-way function? (Prove your answer). If not, is there a way to use f as a building block to build a one-way function? If so, show how to do it and prove your answer; if not, show a counterexample.

**Ans:** The function is not one way. This is because, for a constant fraction ($1/2^{20}$), we can invert $f$ in PPT. We can use $f$ as a building block for a one way function in the following manner by constructing function $f'$. On input $x$, we can return $f(x) \oplus f(\hat{x})$. Only one of the two outputs can have the first 20 bits as 1, so

**Problem 6.** Consider a secure pseudo-random generator (PRG) G that maps n-bit seed s to 2n-bit output G(s). Define LH(*) to be a function that takes as an input 2n bits, and outputs the first half of its input bits. Define RH(*) similarly (for the right half). Define:

$$F(s) = G(LH(G(s)))G(RH(G(s)))$$

1. The output length is $4n$.

2. We demonstrate the security of $G$ by reduction, showing that an adversary that can distinguish a random number from $F(s)$ can differentiate a random number from $G(s)$ with non-negligible probability. This is done using the hybrid argument. We demonstrate that if we have an output that can differentiate between an output of 4 random $n$-bit long numbers and $G(s)$, $G$ is not a secure PRG. Firstly, this implies the existence of an adversary $A$ that outputs 1 if the input is $G(s)$ with non-negligible probability. The idea is the following. First, we replace the left $n$ bits of $G(s)$ with a truly random number. Therefore, the final output of this hybrid machine will be:

$$F_1(S) = G(LH(r_l))G(RH(G(s)))$$

   (a) Firs