1. Suppose you wish to use ChaCha20: \(\{0,1\}^{256} \times \{0,1\}^{128} \rightarrow \{0,1\}^{512}\) to probabilistically encrypt a string \(M\) using the method described in class.\(^1\) How many invocations of ChaCha20 will you need to encrypt a 1024-byte plaintext \(M\)? (A byte is 8 bits.)

2. Let \(F: \{0,1\}^{256} \times \{0,1\}^{128} \rightarrow \{0,1\}^{512}\) be a pseudorandom function (PRF). Using \(F\), define a pseudorandom generator (PRG) \(G: \{0,1\}^{256} \rightarrow \{0,1\}^{512}\). Function \(G\) should be secure (in the PRG-sense) if \(F\) is secure (in the PRF-sense). Make the definition of \(G\) as simple as possible, and make sure your definition is “type correct” (e.g., don’t write an integer where a string is needed).

\[
G(S) = \]

---

\(^1\)A reminder: the first argument is the 256-bit key; the second argument is the 128-bit index. The index is sometimes understood to be partitioned into a nonce and a counter.