18.1 A taxicab was involved in a fatal hit-and-run accident at night. Two cab companies, the Green and the Blue, operate in the city. You are told that

- 85% of the cabs in the city are Green and 15% are Blue.
- A witness identified the cab as Blue.
The court tested the reliability of the witness under the same circumstances that existed on the night of the accident and concluded that the witness was correct in identifying the color of the cab 80% of the time. What is the probability that the cab involved in the incident was Blue rather than Green?

18.2 Assume that passwords are selected from four-character combinations of 26 alphabetic characters. Assume that an adversary is able to attempt passwords at a rate of one per second.
   a. Assuming no feedback to the adversary until each attempt has been completed, what is the expected time to discover the correct password?
   b. Assuming feedback to the adversary flagging an error as each incorrect character is entered, what is the expected time to discover the correct password?

18.3 Assume that source elements of length $k$ is mapped in some uniform fashion into a target elements of length $p$. If each digit can take on one of $r$ values, then the number of source elements is $r^k$ and the number of target elements is the smaller number $r^p$. A particular source element $x_i$ is mapped to a particular target element $y_j$.
   a. What is the probability that the correct source element can be selected by an adversary on one try?
   b. What is the probability that a different source element $x_i (x_i \neq x_k)$ that results in the same target element, $y_j$, could be produced by an adversary?
   c. What is the probability that the correct target element can be produced by an adversary on one try?

18.4 A phonetic password generator picks two segments randomly for each six-letter password. The form of each segment is CVC (consonant, vowel, consonant), where $V = <a, e, i, o, u>$ and $C = V$.
   a. What is the total password population?
   b. What is the probability of an adversary guessing a password correctly?

18.5 Assume that passwords are limited to the use of the 95 printable ASCII characters and that all passwords are 10 characters in length. Assume a password cracker with an encryption rate of 6.4 million encryptions per second. How long will it take to test exhaustively all possible passwords on a UNIX system?

18.6 Because of the known risks of the UNIX password system, the SunOS-4.0 documentation recommends that the password file be removed and replaced with a publicly readable file called /etc/publike.key. An entry in the file for user A consists of a user's identifier $ID_a$, the user's public key, $PU_a$, and the corresponding private key $PR_a$. This private key is encrypted using DES with a key derived from the user's login password $P_a$. When A logs in, the system decrypts $E[P_a, PR_a]$ to obtain $PR_a$.
   a. The system then verifies that $P_a$ was correctly supplied. How?
   b. How can an opponent attack this system?

18.7 The encryption scheme used for UNIX passwords is one way; it is not possible to reverse it. Therefore, would it be accurate to say that this is, in fact, a hash code rather than an encryption of the password?

18.8 It was stated that the inclusion of the salt in the UNIX password scheme increases the difficulty of guessing by a factor of 4096. But the salt is stored in plaintext in the same entry as the corresponding ciphertext password. Therefore, those two characters are known to the attacker and need not be guessed. Why is it asserted that the salt increases security?

18.9 Assuming that you have successfully answered the preceding problem and understand the significance of the salt, here is another question. Wouldn't it be possible to thwart completely all password crackers by dramatically increasing the salt size to, say, 24 or 48 bits?

18.10 Consider the Bloom filter discussed in Section 18.3. Define $k = \text{number of hash functions}; N = \text{number of bits in hash table};$ and $D = \text{number of words in dictionary}$. 
a. Show that the expected number of bits in the hash table that are equal to zero is expressed as

\[ \phi = \left( 1 - \frac{k}{N} \right)^D \]

b. Show that the probability that an input word, not in the dictionary, will be falsely accepted as being in the dictionary is

\[ P = (1 - \phi)^k \]

c. Show that the preceding expression can be approximated as

\[ P \approx (1 - e^{-kD/N})^k \]

18.11 Design a file access system to allow certain users read and write access to a file, depending on authorization set up by the system. The instructions should be of the format

**READ** \(F, \text{User } A\): attempt by User A to read file F

**WRITE** \(F, \text{User } A\): attempt by User A to store a possibly modified copy of F

Each file has a **header record**, which contains authorization privileges; that is, a list of users who can read and write. The file is to be encrypted by a key that is not shared by the users but known only to the system.