1. Suppose you flip 1000 pennies (worth one cent each) and 1000 dimes (worth ten cents each). You get to keep the ones that land heads up.
   a. What is the expected value of the coins you keep?
   b. What is the variance of that value?

2 You have items of Types 1 and 2. Each item is of either Type 1 or Type 2, but you can not directly determine its type. Each item has three properties: A, B, and C. Each property has one of two values: + and −. The following table gives the probability that an item is of Type 1 depending on the value of single property.

	+	−
A	0.9	0.1
B	0.8	0.2
C	0.7	0.3

Assume that the properties are statistically independent.
   a. If an item is of Type 1, how likely is it that all of the following will be true: A is +, B is +, and C is +?
   b. If an item is of Type 1, how likely is it that at least one of the following will be true: A is +, B is +, or C is +?
   c. Suppose we say an item is in Class 1 when both A and B have the value +. What fraction of the items in Class 1 are of Type 1?
   c. Suppose we say an item is in class 1 when at least two of A, B, and C have the value +. What fraction of the items in Class 1 are of Type 1?

3. Suppose Algorithm 1.8 is modified to multiply an m digit number by an n digit number, where each digit occupies one computer word.
   a. As a function of m and n, how many one digit numbers are multiplied by the algorithm? (All of these multiplications are done in Step 5 of the algorithm.)
   b. How many times is a two digit number, a one digit number and a carry added? (All of these additions are done in Step 5.)

4. Suppose Algorithm 1.8 is modified to multiply an m/s digit number by an n/s digit number, where each digit needs s words. (Also assume that m/s and n/s are integers.)
   a. How many s word numbers are multiplied by the algorithm?
   b. How many times is a $2s$ word number, a s word number and a carry added.

5. Suppose an m-word number is multiplied by an n-word number words using the algorithm of Question 4, where that algorithm calls the algorithm of Question 3 to do its multiplications of s word numbers. (Assume that m and n are multiplies of s.)
   a. How many times will this combined algorithm multiply one word numbers? (Your answer might depend on m, n, and s.)
   b. How many times will this combined algorithm perform the additions referred to in Question 3b?
   c. How many times will this combined algorithm perform the additions referred to in Question 4b?
   d. Let t_a be the time needed to multiply one word numbers, t_b be the time needed to do the additions referred to in Question 3b, and t_c be the time needed to do the additions referred to in Question 4b. For this subquestion write three formulas:
   (1) the time needed to multiply an m word number by an n word number using the algorithm of Question 3 directly,
   (2) the time needed to multiply the same numbers using the algorithm of this question,
   (3) the result of subtracting the time form part 3d(1) from the time from 3d(2).
   Finally, say which algorithm is faster, or discuss the situation if it is not completely clear which method is faster. (To keep the question from being overly complex, this question is assuming that no time is needed for anything other than the indicated additions and multiplications. As an exercise at home, you might want to see whether it makes any important differences if you include all the reasons why each algorithm uses time.)