Illinois Institute of Technology
Department of Computer Science

Solutions to Homework Assignment 3

CS 330 Discrete Structures
Fall Semester, 2009

1. Page 415, problem 18.

   (a) What is the probability that two people chosen at random were born on the same day of the week?
   1/7

   (b) What is the probability that in a group of n people, there were at least two born on the same day of the week?

   $$1 - \frac{6}{7} \cdot \frac{5}{7} \cdots \frac{8-n}{7}$$

   (c) How many people chosen at random are needed to make the probability greater than 1/2 that there are at least two people born on the same day of the week?

   4 people.

2. Page 416, problem 30. Find the probability that a randomly generated bit string of length 10 does not contain a 0 if bits are independant and if

   (a) a 0 bit and a 1 bit are equally likely
   0.5^{10}

   (b) the probability that a bit is a 1 is 0.6.
   0.6^{10}

   (c) the probability that the ith bit is a 1 is $1/2^i$ for $i = 1, 2, 3, \ldots, 10$

   $$\prod_{i=1}^{10} \left(\frac{1}{2}\right)^i = \frac{1}{2}^{\sum_{i=1}^{10} i} = \left(\frac{1}{2}\right)^{55}$$

3. Page 425, problem 22. Suppose that we have prior information concering whether a random incoming message is spam. In particular, suppos that over a time period we find that s spam messages arrive and h messages arrive that are not spam.

   (a) Use this information to estimate $p(S)$ the probability that an incoming message is spam, and $p(\bar{S})$, the probability an incoming message is not spam.
   $p(S) = \frac{s}{s+h}$ and $p(\bar{S}) = \frac{h}{s+h}$

   (b) Use Bayes' theorem and part (a) to estimate the probability that an incoming message containing the word w is a spam, where $z(w)$ is the probability that w appears in a spam message, and $q(w)$ is the probability that w appears in a message that is not spam. (I have changed $p(w)$ in the book here to $z(w)$ to avoid confusion with the conditional probabilities I will use below.)
   In other words, $p(w|S) = z(w)$ and $p(w|\bar{S}) = q(w)$. So $p(w) = p(w|S)p(S) + p(w|\bar{S})p(\bar{S})$
   By bayes theorem, $p(S|w) = \frac{p(w|S)p(S)}{p(w)}$ so we can substitute quantities that we have computed so far:

$$\begin{aligned} p(S|w) &= \frac{p(w|S)p(S)}{p(w)} \\ &= \frac{p(w|S)P(S)}{p(w|S)p(S) + p(w|\bar{S})p(\bar{S})} \\ &= \frac{z(w)\frac{s}{s+h}}{z(w)\frac{s}{s+h} + q(w)\frac{h}{s+h}} \\ &= \frac{sz(w)}{sz(w) + hq(w)} \end{aligned}$$

In the book's notation for this problem, this is

$$\frac{sp(w)}{sp(w) + hq(w)}$$

4. Page 441, problem 34. Suppose the probability that x is the ith element in a list of n distinct integers is $\frac{i}{n(n+1)}$. Find the average number of comparisons used by the linear search algorithm to find x or determine that it is not in the list.

The average number of comparisons needed to find x is

$$\text{(maximum number of comparisons)}P(\text{not finding } x \text{ at all}) + \\ \sum_i \text{(number of comparisons)}P(\text{finding } x \text{ in position } i)$$

Ordinarily we would also include

$$P(\text{not finding } x \text{ before position } i)$$

but here we don't need to do that because we know the integers are distinct, so if we find x at position i, there is no chance of finding another copy of x at a position before i.

The number of comparisons is just the number of array elements we check. We can ignore any extra comparisons that might be required to check array bounds because they're not comparisons against array elements. It's generally useful to make this distinction because comparisons against array elements can be much expensive, than the array bounds checks. For example when the array elements are long strings you need to check every character in both strings (worst case), whereas checking an array index is an integer comparison which requires only a single processor instruction. If you included bounds checks, you did not lose any points.

We note that $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$, so the probability of finding x in the list is $\frac{1}{2}$, as is the probability that x is not in the list.

Thus, the average number of comparisons is

$$\frac{n}{2} + \sum_{i=1}^{n} i \cdot \frac{i}{n(n+1)} = \frac{n}{2} + \frac{n(n+1)(2n+1)}{6n(n+1)} = \frac{3n}{6} + \frac{2n+1}{6} = \frac{5n+1}{6}$$

If you included bounds checks, you may have instead set up the equation as follows, depending on the exact set of comparisons in your algorithm.

$$\frac{2n+2}{2} + \sum_{i=1}^{n}(2i+1) \cdot \frac{i}{n(n+1)}$$

In this case the solution is

$$n+1+2\cdot\frac{n(n+1)(2n+1)}{6n(n+1)}+n=2n+1+2\cdot\frac{2n+1}{6}=\frac{6n+3}{3}+\frac{2n+1}{3}=\frac{8n+4}{3}$$

5. Page 444, problem 16. What is the probability that when a fair coin is flipped n times, an equal number of heads and tails appear?

   This problem is most easily solved using the binomial distribution. Look it up and plug in the numbers.

$$\begin{array}{cl}\binom{n}{n/2}\cdot 0.5^n & \text{if } n \text{ is even}\\ 0 & \text{if } n \text{ is odd}\end{array}$$

   Other people answered as follows:

$$P=\frac{E_1}{E_2}$$

   E_1 is the number of events where there are $n/2$ heads and $n/2$ tails (which is $\binom{n}{n/2}$ if n is even, and 0 if n is odd).

   E_2 is the total number of arrangements of heads and tails which is 2^n.