CS 1050B: Constructing Proofs

Quiz 2

1. (10 points) Find the solution to each of these recurrence relations and initial conditions.

   a) $a_n = a_{n-1} + n, a_0 = 1$

   b) $a_n = 2na_{n-1}, a_0 = 1$

   Solution :

   a)

$$\begin{aligned} a_n &= n + a_{n-1} \\ &= n + ((n-1) + a_{n-2}) \\ &= (n + (n-1)) + ((n-2) + a_{n-3}) \\ &= (n + (n-1) + (n-2) + \cdots + 1) + a_0 \\ &= \frac{n^2 + n + 2}{2} \end{aligned}$$

   b)

$$\begin{aligned} a_n &= 2na_{n-1} \\ &= 2n(2(n-1)a_{n-2}) \\ &= 2^2(n(n-1))a_{n-2} \\ &= 2^n(n(n-1)(n-2)\cdots(n-(n-1)))a_{n-n} \\ &= 2^n n! \end{aligned}$$

2. (10 points) A bowl contains 10 red balls and 10 blue balls. A woman selects balls at random without looking at them.

   a) How many balls must she select to be sure of having at least three balls of the same color?

   b) How many balls must she select to be sure of having at least three blue balls?

   Solution :

   a) we have $p = 2$ bins (pigeonholes) and $r = 3$: the minimum number of balls (pigeons) in any of the bins. By the generalized pigeonhole principle, the answer is $p(r-1)+1 = 5$.

   b) The answer is 13.

3. (10 points) A long straight. (You don't need to work out the actual numbers for this problem.)

   a) How many ways are there to pick out 13 cards in a 52-card deck?
   b) How many 13-card hands are there that contains no pairs?
   c) What is the probability that a 13-card hand contains no pairs?

   Solution :

   a) $C(52, 13)$
   b) A hand with no pairs must contain exactly one card of each kind. The only choice involved, therefore, is the suit for each of the 13 cards. There are 4 ways to specify the suit, and there are 13 tasks to be performed. Therefore, there are 4^{13} hands with no pairs.
   c) $4^{13}/C(52, 13)$

4. (10 points) The merge sort algorithm splits a list to be sorted with n items, where n is even, into two lists with $n/2$ elements each, and uses fewer than n comparisons to merge the two sorted lists of $n/2$ items each into one sorted list.

   a) Set up a divide-and-conquer recurrence relation $f(n)$ for the number of comparisons used by the merge sort to sort a list of n elements.
   b) Find a big-O estimate for $f(n)$.

   Solution :

   a) $f(n) = 2f(n/2) + n$
   b) By the Master Theorem, with $a = 2, b = 2, c = 1$, and $d = 1$, we see that $f(n) = O(n \log n)$.

5. (10 points) A special die is biased so that a 3 comes up twice as often as each other number.

   a) What is the probability to roll a 3? What about the other numbers?
   b) What is the expected value of the die?
   c) What is the variance? (You don't need to work out the actual numbers.)

   Solution :

   a) $p(3) = 2/7$ and $p(1) = p(2) = p(4) = p(5) = p(6) = 1/7$.
   b) $E(X) = 2/7 \cdot 3 + 1/7 \cdot (1 + 2 + 4 + 5 + 6) = 24/7$

c)

$$\begin{aligned} V(X) &= E(X^2) - E(X)^2 \\ &= (2/7 \cdot 3^2 + 1/7 \cdot (1^2 + 2^2 + 4^2 + 5^2 + 6^2)) + (24/7)^2 \end{aligned}$$