Practice Questions for Feb 19, 2019

1. We showed that we can form a committee of a president, a secretary, and 3 other members out of 10 people in 10×9×C(8, 3) = 5040 ways. This was achieved by doing the following steps:

   (P) choose the president first (in 10 ways, out of 10 people), then

   (S) choose the secretary (in 9 ways, out of 9 remaining people), and then

   (O) choose the other 3 members (in C(8, 3) = 56 ways, out of remaining 8 people).

   We can call this the ⟨P, S, O⟩ approach.

   One could do the steps P, S, and O in some other order. For example, O first, then S, and then P; We can call this the ⟨O, S, P⟩ approach.

   Show the number of ways to do the step O first, the number of ways to do S next, and the number of ways to do P next. What is the number of ways to form the committee now using the ⟨O, S, P⟩ approach?

   Repeat the above now for all other approaches.

2. Consider the code below (it is the same code we considered in the short-quiz on Feb 14).

```
1.  for (int i=0; i<H.length; i++)
2.      if (H[i] > W[i]) return(false);
3.  return(true);
```

   The table below (except the last row) shows the tests done in line 2 for different loop-iterations for H = [1, 1, 0, 1] and the kind of W[] that could make the code exit with the return value **false** in any given iteration. Notes: (1) In col. 5, all previous tests in line 2 must fail (to avoid exiting the loop); in col. 6, the most recent test must succeed to exit the loop. (2) The last row in the table is not an iteration of the loop, and it only has the 2nd column and the last two columns. (3) All W[] types in the last column are disjoint, i.e., no W[] fits more than one type; also, each of 2^4 = 16 W[]'s fits one type.

Iteration	i	Test in line 2 in this iteration	W[i] to satisfy the test and exit loop	All previous tests in line 2 and their successes(T)/failures(F)	What we know about W[] if loop exits now and type of W[]
1	0	H[0] > W[0], i.e., 1 > W[0]	W[0] = 0	None	W[0] = 0; W = [0, ?, ?, ?]
2	1	H[1] > W[1], i.e., 1 > W[1]	W[1] = 0	1 > W[0] (F)	W[0] = 1, W[1] = 0; W = [1, 0, ?, ?]
3	2	H[2] > W[2], i.e., 0 > W[2]	No W[2]; cannot exit	1 > W[0] (F), 1 > W[1] (F)	W[0] = 1, W[1] = 1, W[2] = ?; cannot exit
4	3	H[3] > W[3], i.e., 1 > W[3]	W[3] = 0	1 > W[0] (F), 1 > W[1] (F), 0 > W[2] (F)	W[0] = 1, W[1] = 1, W[2] = ?, W[3] = 0; W = [1, 1, ?, 0]
	4			1 > W[0] (F), 1 > W[1] (F), 0 > W[2] (F), 1 > W[3] (F)	W[0] = 1, W[1] = 1, W[2] = ?, W[3] = 1; W = [1, 1, ?, 1]

3. Repeat Problem 2 for the following H = [0, 1, 0, 0, 1, 1]; also show computation of average #(iterations). Note that #(iterations) for the case of return value **true** is H.length.

4. Repeat Problem 2 for the following code (the loop is now iterated in decreasing values of i) and H = [1, 0, 0, 0, 1, 1]; also show computation of average #(iterations).

```
1.  for (int i=H.length-1; i>=0; i--)
2.      if (H[i] > W[i]) return(false);
3.  return(true);
```

5. Use the formula $C(n, m) = n!/(m!(n-m)!)$ to show $C(m+n+p, m)C(n+p, n) = C(m+n+p, n)C(m+p, m)$. Give two other such equalities.