Practice Questions for Jan 31, 2019

1. To compute $C(n, m)$ using the formula $C(n, m) = \frac{n}{m} C(n-1, m-1)$, assuming $m \le n/2$, we succesively compute $N_1 = \frac{n-m+1}{1} = n-m+1$, $N_2 = (n-m+2) * N_1/2$, $N_3 = (n-m+3)N_2/3$, and so on, finally $N_m = n * N_{m-1}/m = C(n, m)$.

   (a) Give a for-loop to carry out this computation of $C(n, m)$; use the integer-variable name "result" for the successive values $N_1, N_2, \cdots, N_m$ computed.

   (b) Express the number of multiplications and divisions done in terms of n and m.

   (c) Repeat (a)-(b) for computing $C(n, m)$ based on the formula $C(n, m) = \frac{n-m+1}{m} C(n, m-1)$.

   (e) Is there any advantage of the for-loop in (a) vs. that in (c) in terms of #(iterations), #(multiplication and division operations), and the nature of intermediate values (being large vs. small) of result? Justify your answer with examples.

2. We know a subset can be represented by an array of 1's and 0's (1 for an item in the subset and 0 for an item not in the subset). Shown below are some example subsets of {A, B, C, D, E} and their array-representation. Note that #(1's in the array) equals the size of the subset.

Subsets	A	B	C	D	E	Array representation
$\{A, B, E\}$	1	1	0	0	1	[1, 1, 0, 0, 1]
$\{B\}$	0	1	0	0	0	[0, 1, 0, 0, 0]
$\varnothing$	0	0	0	0	0	[0, 0, 0, 0, 0]

   What does the following code return when subsetOne and subsetTwo equal the arrays for subsets $\{A, B, E\}$ and $\{B\}$?

```
for (int i=0; i<subsetOne.length; i++)
    if (subsetOne[i] != subsetTwo[i]) return (false);
return (true);
```

   How many times the comparison "subsetOne[i] != subsetTwo[i]" is done?

   State in English what the code returns, in general, for two input arrays subsetOne and subsetTwo (of 0's and 1's and having the same length)?

   For subsetOne = [1, 1, 0, 0, 1], how many subsetTwo give the return value "true" and what is #(comparison "subsetOne[i] != subsetTwo[i]") for each of those subsetTwo?

   For subsetOne = [1, 1, 0, 0, 1], how many subsetTwo give the return value "false" and how many of them have #(comparison "subsetOne[i] != subsetTwo[i]") = k for each of k = 1, 2, 3, and 4?

   Give the codes to test each of the following: (a) subsetOne is a subset of subsetTwo, and (b) subsetOne and subsetTwo are disjoint.

3. We know that 4 straightlines L1, L2, L3, and L4 can intersect in minimum 0 points and maximum 6 = C(4,2) points. Show whether it is possible for 4 straightlines to intersect in exactly k points for k = 1, 2, 3, 4, and 5. (Note that n straightlines L1, L2, ..., Ln can intersect in maximum C(n,2) = n(n-1)/2 points.)