Name ______________________

CS200 - Midterm 2 Review Questions

1. Suppose you are creating a password that is 6 characters long, using numbers, upper case letters, and lower case letters. How many passwords are possible, if you want to use 2 numbers, 2 upper case letters, and 2 lower case letters?

   Solution We will break this problem up into subtasks, and then use the product rule to count the total number of options. The first task is to choose the two positions out of 6 where we will put the 2 numbers. There are $\binom{6}{2}$ ways to do this. Now that we've decided where to put the numbers, we need to decide which numbers to choose. We have 10 choices for the first number, and 10 choices for the second number, giving us 100 options. Next we need to decide where to put the 2 upper case letters. Since 2 spots are already taken up with numbers, we have four spots to choose from, so there are $\binom{4}{2}$ ways to choose where to put the upper case letters. Now that we've decided where to put the upper case letters, we need to decide which letters to choose. We have 26 options for the first letter and 26 options for the second, for 26^2 total options. Finally, the two lower case letters must go in the two remaining spaces, so we don't have any further options in terms of positions. We do get to choose which lower case letters to use, which gives us another 26^2 possibilities. All together, we have $\binom{6}{2} \times \binom{4}{2} \times 10^2 \times 26^4$.

2. For many more good practice problems with solutions involving counting, see DMOI Counting Chapter Review.

3. [**11 points**] Find a value n_0, and then prove that for all integers $n \geq n_0$, it is possible to create n-cents worth of postage out of 4-cent stamps and 9-cent stamps. Do not use regular induction. Your n_0 does not need to be the smallest number possible, just any number that works.

   Solution Let $P(n)$ be the predicate that n cents of postage can be created from 4- and 9- cent stamps. We will prove $P(n)$ is true for all $n \geq 24$ using strong induction.

   Base case: $P(24)$ is true because $6 \times 4 = 24$. $P(25)$ is true because $9 + 4 \times 4 = 25$. $P(26)$ is true because $9 \times 2 + 4 \times 2 = 26$. $P(27)$ is true because $3 \times 9 = 27$.

   Inductive step: Let $k \geq 27$, and assume for strong induction that $P(j)$ is true for all $k \geq j \geq 24$. Then since $k + 1 \geq 28$, $k + 1 - 4 \geq 24$, so $P(k + 1 - 4)$ is true. This means there is a way of creating $k + 1 - 4$ cents of postage out of 4 and 9 cent stamps. But then we can just add a 4-cent stamp to this solution to get a way of creating $k + 1$ cents using 4 and 9-cent stamps. Thus $P(k + 1)$ is true.

   By strong induction, we've shown $P(n)$ is true for all $n \geq 24$.

4. (Your pseudocode will be graded on correctness, readability, elegance, and appropriate documentation.) Create pseudocode that takes as input a directed graph $G = (V, E)$ in either adjacency matrix or adjacency list form, and tests whether the graph has any self-loops.

Algorithm 1: `AdListFunc`(A)

Input : Adjacency List A of a graph $G = (V, E)$.
Output: True if G has any self-loops, false otherwise.

Algorithm 2: `AdListMat`(A)

Input : Adjacency Matrix A of a graph $G = (V, E)$..
Output: True if G has any self-loops, false otherwise.

Solution

Algorithm 3: `AdListFunc`(A)

```
Input  : Adjacency List A of a graph G = (V, E).
Output: True if G has any self-loops, False otherwise.
for v ∈ V do
    for u ∈ A[v] do
        if u = v then
            Return True;
        end
        // We check whether v is an element of the adjacency list of v.
    end
end
// We only get to this point if we have not seen any self-loops
Return False;
```

Algorithm 4: `AdListMat`(A)

```
Input  : Adjacency Matrix A of a graph G = (V, E).
Output: True if G has any self-loops, false otherwise.
for v ∈ V do
    if A[v, v] = 1 then
        Return True;
        // Checking that there is a self-loop at vertex v.
    end
end
// We only get to this point if we have not seen any self-loops
Return False;
```

5. Let $f : \mathbb{R} \to \mathbb{R}$ be defined as $f(x) = \lfloor x \rfloor$. Explain why $f(x) = \Theta(x)$.

   Solution We need to show $f(x) = O(x)$ and $f(x) = \Omega(x)$. But $f(x) = O(x)$ for $C = 1$ and $k = 0$, and $f(x) = \Omega(x)$ for $C = 1/2$ and $k = 1$. Thus $f(x) = \Theta(x)$.

6. Lots more practice problems with big-O in the Rosen chapter on Canvas ("Function Growth.")

7. For more strong induction practice, see DMOI Induction Excercises especially 19, 20.

8. For more "story" (non-mathy) proofs, see DMOI proof and induction chapter exercises.