Homework 2
Show all work.

1. For each of the following pairs of functions $f$ and $g$, state whether $f$ is $o(g)$, $\Theta(g)$, or $\omega(g)$, with a proof.

(a) $f = \sqrt{n}, g = \lg(n)$
Both $\lim_{n \to \infty} f(n) = \infty$, and $\lim_{n \to \infty} g(n) = \infty$, and $f$ and $g$ are differentiable, so we apply L’Hopital’s rule:

\[
f'(n) = \frac{1}{2\sqrt{n}} \quad \text{and} \quad g'(n) = \frac{1}{\ln n} \quad (\text{we used the identity that } \lg(n) = \frac{\ln n}{\ln 2})
\]

so

\[
\lim_{n \to \infty} \frac{f'(n)}{g'(n)} = \lim_{n \to \infty} \frac{\ln 2 \cdot n}{2\sqrt{n}} = \lim_{n \to \infty} \frac{\ln 2 \sqrt{n}}{2} = \infty
\]

Thus, $f$ is $o(g)$.

(b) $f = n \cdot \lg(n), g = n^2$

Both $\lim_{n \to \infty} f(n) = \infty$, and $\lim_{n \to \infty} f(n) = \infty$, and $f$ and $g$ are differentiable, so we apply L’Hopital’s rule. We first simplify the ratio $\frac{f(n)}{g(n)}$:

\[
\frac{f(n)}{g(n)} = \frac{n \cdot \lg(n)}{n^2} = \frac{\lg(n)}{n}
\]

Now we take the derivatives $\frac{\partial}{\partial n} \lg(n) = \frac{1}{\ln n}$ and $\frac{\partial}{\partial n} n = 1$, and take the limit

\[
\lim_{n \to \infty} \frac{1}{\ln 2 \cdot n} = 0
\]

Thus, $f$ is $o(g)$.

(c) $f = n!, g = n^n$ It is easy to figure out but hard to prove that $n^n$ is the dominating function so $f$ is $o(g)$. The reason it is difficult is that we don’t know how to take the derivative of $n!$. Therefore, we must either go back to the basic definition of little-o or bound $n!$ using Stirling’s formula to get a differentiable function. We give the first of these methods.

To show that $n!$ is $O(n^n)$, we must show that for any $c > 0$, there is an $N$ such that $c \cdot n! < n^n$ for all $n > N$. In other words, given any $c$, we want to find a large enough $N$ so the inequality holds. Let $N = c^2$. Then we will show that $c \cdot n! < n^n$ when $n > N$. First, we show that $c \cdot (c^2)! < (c^2)!(c^3)$:

\[c \cdot (c^2)! = c \cdot c^2 \cdot (c^2 - 1) \cdot (c^2 - 2) \cdots 3 \cdot 2 \cdot 1\]
\[ = c \cdot c^2 \cdot (c^2 - 1) \cdot (c^2 - 2) \ldots 3 \cdot 2 \text{ which is a product of } c^2 \text{ terms, none of which is greater than } c^2 \text{ and some of which are less than } c^2. \text{ And } (c^2)^2 \text{ is a product of } c^2 \text{ terms all of which equal } c^2, \text{ so } c \cdot (c^2)! < (c^2)!^2. \] Now we show that this inequality still holds for all \( n > c^2 \): Suppose \( c \cdot u! < u^n \). Then \( c \cdot (u+1)! = c \cdot (u+1)u! < (u+1)u^n \) (using the hypothesis that \( c \cdot u! < u^n \)) \( < (u+1)(u+1)^n = (u+1)^{u+1} \), which establishes the result by induction. Therefore, for any constant \( c \), we have \( c \cdot n! < n^n \) if \( n > c^2 \), which shows by the definition that \( n! \text{ is } o(n^n) \).

(d) \( f = 2^n, g = 2^{n+1} \)

\[
\lim_{n \to \infty} \frac{f(n)}{g(n)} = \lim_{n \to \infty} \frac{2^n}{2^{n+1}} = \frac{1}{2}
\]

Thus, \( f(n) \) is \( \theta(g(n)) \).

(e) \( f = n^{1000}, g = 2^n \)

Let \( f^{(1000)} \) be the thousandth derivative of \( f \), which equals 1000! (since \( f' = 1000 \cdot n^{999}, f'' = 1000 \cdot 999 \cdot n^{998}, \ldots, f^{(k)} = 1000 \cdot 999 \ldots (k + 2) \cdot (k + 1) \cdot k \).

Similarly \( g^{(1000)} = (\ln 2)^{1000} \cdot 2^n = c \cdot 2^n \) (since \( g' = \ln 2 \cdot 2^n, g'' = (\ln 2)^2 \cdot 2^n, g^{(k)} = (\ln 2)^k \cdot 2^n \)). Thus,

\[
\lim_{n \to \infty} \frac{f(n)}{g(n)} = \lim_{n \to \infty} \frac{f^{(1000)}(n)}{g^{(1000)}(n)} = \lim_{n \to \infty} \frac{1000!}{(\ln 2)^{1000} \cdot 2^n} = \lim_{n \to \infty} \frac{c}{2^n} \text{ where } c = \frac{1000!}{(\ln 2)^{1000}} = 0
\]

Thus \( f \) is \( o(g) \).

2. Prove: If \( P(n) \) is a polynomial in \( n \), then \( \log(P(n)) \) is \( \theta(\log(n)) \).

Proof: Let \( P(n) \) be a polynomial of degree \( k \), i.e. \( P(n) = a_1 n^k + a_2 n^{k-1} + \ldots + a_k n + a_{k+1} \). First, we show that \( P(n) \) is \( \theta(n^k) \):

\[
\lim_{n \to \infty} \frac{P(n)}{n^k} = \lim_{n \to \infty} \frac{a_1 n^k + a_2 n^{k-1} + \ldots + a_k n + a_{k+1}}{n^k} = \lim_{n \to \infty} a_1 + \frac{a_2}{n} + \ldots + \frac{a_k}{n^{k-1}} + \frac{a_{k+1}}{n^k} = a_1
\]

so \( P(n) \) is \( \theta(n^k) \).
Now, \( \log(n^k) = k \log(n) \) so \( \log(n^k) \) is \( \Theta(\log(n)) \), since they bound each other by a constant factor. Putting these together, \( \log(P(n)) = \Theta(\log(n)) \)

3. Prove that \( \binom{n}{k} \) is \( O(n^k) \), where \( k \) is a constant and \( \binom{n}{k} \) represents the \( k \)th binomial coefficient, whose formula is \( \frac{n!}{k!(n-k)!} \).

We have \( \binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)(n-2)\ldots(n-(k+2))(n-(k+1))(n-k)!}{k!} \) which is the constant \( \frac{1}{k!} \) times a product of the \( k \) terms \( n, (n-1), (n-2) \ldots (n-(k+1)) \) each of which is at most \( n \), so their product is at most \( n^k \). Thus, let \( c = k! \) and we have \( c \cdot \binom{n}{k} \leq n^k \) and therefore \( \binom{n}{k} \) is \( O(n^k) \).

4. You roll two dice. The dice are standard cubes with the numbers 1 through 6 on the faces and are fairly weighted so that each face is equally likely to come up. What is the expected value of the sum of the numbers on the dice faces?

Answer: Let \( P_i \) denote the probability of \( i \) being the sum of the dice. There are 36 possible combinations of rolls of the two dice (6 possible faces on the first times 6 possible faces on the second). We compute the probabilities by enumerating cases:

\[
\begin{align*}
P_2 & = \frac{1}{36} \text{ (roll 1-1)} \\
P_3 & = \frac{2}{36} \text{ (roll 1-2, or 2-1)} \\
P_4 & = \frac{3}{36} \text{ (roll 1-3, 2-2, or 3-1)} \\
P_5 & = \frac{4}{36} \text{ (roll 1-4, 2-3, 3-2, or 4-1)} \\
P_6 & = \frac{5}{36} \text{ (roll 1-5, 2-4, 3-3, 4-2 or 5-1)} \\
P_7 & = \frac{6}{36} \text{ (roll 1-6, 2-5, 3-4, 4-3, 5-2 or 6-1)} \\
P_8 & = \frac{5}{36} \text{ (roll 2-6, 3-5, 4-4, 5-3 or 6-2)} \\
P_9 & = \frac{4}{36} \text{ (roll 3-6, 4-5, 5-4 or 6-3)} \\
P_{10} & = \frac{3}{36} \text{ (roll 4-6, 5-5 or 6-4)} \\
P_{11} & = \frac{2}{36} \text{ (roll 5-6 or 6-5)} \\
P_{12} & = \frac{1}{36} \text{ (roll 6-6)}
\end{align*}
\]

\[
\mathcal{E}(\text{sum of dice}) = \sum_{i=2}^{12} i P_i
\]

\[
= 2 \cdot \frac{1}{36} + 3 \cdot \frac{2}{36} + 4 \cdot \frac{3}{36} + 5 \cdot \frac{4}{36} + 6 \cdot \frac{5}{36} + 7 \cdot \frac{6}{36}
+ 8 \cdot \frac{5}{36} + 9 \cdot \frac{4}{36} + 10 \cdot \frac{3}{36} + 11 \cdot \frac{2}{36} + 12 \cdot \frac{1}{36}
\]

\[
= 7
\]

Alternate solution: We first compute the expected value of the face value of one dice. The probability of each face is \( 1/6 \) by the assumption that each is equally likely, and the possible values are from 1 to 6. Thus, the
expected value is \( \sum_{i=1}^{6} i \cdot 1/6 = \frac{6 \cdot \frac{7}{2}}{2} = 3.5 \). The expected value of the sum of two random variables is the sum of the expected values of each one (see section in text on expected value), so the expected sum for two dice is \( 3.5 + 3.5 = 7 \).

5. **Extra Credit:**

   **Part 1:**
   You have \( n \) puzzle pieces in a jigsaw puzzle. You ignore all pictures and search blindly for pieces that fit together, using the following algorithm:

   (a) Select any piece from unused pile. Call this your partial solution.

   (b) Pick a piece in your partial solution with an open edge. Try each remaining piece in unused pile to see if it fits with this edge of this piece.

   (c) When you find a piece that fits, remove it from the unused pile and add it to the partial solution.

   (d) If the puzzle is not yet complete, return to step 2 above.

   Give a big-\( \Theta \) approximation for the number of times step 2 will happen in the worst case, and explain your reasoning.

   **Solution:** On the first search, we try at worst \( n - 1 \) pieces until we find a fit and add it to the partial solution. On the second search, we try at worst \( n - 2 \) pieces. On the \( k \)-th search, we try \( n - k \) pieces. After \( n - 1 \) searches, the puzzle will be completed. Therefore, a summation for the number of steps is

   \[
   \sum_{i=1}^{n-1} (n - i) = (n - 1)n - \frac{(n - 1)(n)}{2} = \frac{(n - 1)n}{2} = n^2/2 - n/2
   \]

   which is \( \Theta(n^2) \).

   **Part 2:**
   Now suppose somebody has sorted the puzzle pieces and gives you four separate piles of \( n/4 \) puzzle pieces each, and these piles are the four quarters of the puzzle. You use the same algorithm as above to solve each of the four smaller puzzles, then in the end you piece together the four quarters in three final steps. Is there any improvement in either the big-\( \Theta \) expression or in constant factors? Explain your reasoning.

   **Solution:** In this case we have 4 subproblems of size \( n/4 \). Since the algorithm scales as \( n^2 \), a problem of size \( n/4 \) will take steps scaling as \( (n/4)^2 \) or \( n^2/16 \). Then the time to complete 4 of them will scale as \( 4 \cdot n^2/16 \) or \( n^2/4 \). Thus, there would be a speedup by a constant factor of 4.

   You could also determine this exactly: \( 4 \cdot [(n/4)^2/2 - (n/4)/2] + 3 = 4 \cdot (n^2/32 - n/8) + 3 = n^2/8 - n/2 + 3 \), which is indeed faster by a factor of 4 in the leading term of \( n^2/8 \) versus the old leading term of \( n^2/2 \) (as we know, in a polynomial the leading term dominates asymptotically).