INSTRUCTIONS:

1. Use the provided latex template to write your solutions.
2. Submit your solution to Gradescope. Make sure only one of your group members submits. After submitting, make sure to add your group members.

Problem 1 (15 points).

For each pairs of functions below, indicate one of the three: $f = O(g)$, $f = \Omega(g)$, or $f = \Theta(g)$.

1. $f(n) = n^{1.01}$, $g(n) = n^{0.99} \cdot \log n$
2. $f(n) = n$, $g(n) = n^{0.99} \cdot (\log n)^2$
3. $f(n) = 100 \cdot \log(100 \cdot n^3)$, $g(n) = (\log n)^2$
4. $f(n) = n^2 \cdot \log n^2$, $g(n) = n \cdot (\log n)^3$
5. $f(n) = n^{2.01} \cdot \log n^3$, $g(n) = n^2 \cdot (\log n)^2$
6. $f(n) = (\log n)^{\log n}$, $g(n) = n^2$
7. $f(n) = (\log n)^{\log n}$, $g(n) = n^{\log \log n}$
8. $f(n) = (n \cdot \log n)^{\log n}$, $g(n) = n^{(\log n)^2}$
9. $f(n) = n^2 \cdot 2^n$, $g(n) = 3^n$
10. $f(n) = 2^{n \cdot \log n}$, $g(n) = 3^n$
11. $f(n) = n!$, $g(n) = (\log n)^n$
12. $f(n) = n!$, $g(n) = n^n$.
13. $f(n) = \log(n!)$, $g(n) = \log n^n$.
14. $f(n) = \sum_{k=1}^{n} (1/k)$, $g(n) = \log n$
15. $f(n) = \sum_{k=1}^{n} k^2$, $g(n) = n^2 \cdot \log n$

Problem 2 (10 points).

Solve each of the following recursions using master theorem.

1. $T(n) = 16 \cdot T(n/2) + 100 \cdot n^2$.
2. $T(n) = 4 \cdot T(n/2) + 1000 \cdot n^2$.
3. $T(n) = 8 \cdot T(n/2) + 10 \cdot n^{3.5}$.
4. $T(n) = 2 \cdot T(n/2) + n \cdot \log(n)$.
5. $T(n) = 8 \cdot T(n/2) + n^{1.5} \cdot \log^2(n)$.

Problem 3 (9 points).

Assume you have functions f and g satisfying that $f(n)$ is $O(g(n))$. For each of the following statements, decide if you think it is true or false and give a proof or counterexample.

1. $\ln f(n) = O(\ln g(n))$.
2. $2^{f(n)} = O(2^{g(n)})$.
3. $f(n)^2 = O(g(n)^2)$.

Problem 4 (10 points).

Consider recursion $T(n) = \Theta(n) + T(a \cdot n) + T(b \cdot n)$, $T(1) = 1$, $a \geq 0$, $b \geq 0$. Prove the following:

1. $T(n) = \Theta(n)$, if $a + b < 1$.
2. $T(n) = \Theta(n \cdot \log n)$, if $a + b = 1$.

Problem 5 (16 points).

For each pseudo-code below, give the asymptotic running time in Θ notation. You may assume that standard arithmetic operations take $\Theta(1)$ time.

1.
```
for i := 1 to n do
    j := i;
    while j < n do
        j := j + 5;
    end
end
```

2.
```
for i := 1 to n do
    for j := 4i to n do
        s := s + 2;
    end
end
```

3.
```
for i := 1 to n do
    j := n;
    while i^5 < j do
        j := j - 1;
    end
end
```

4.
```
for i := 1 to n do
    j := 2;
    while j < i do
        j := j^4;
    end
end
```

Problem 6 (10 points).

Design a divide-and-conquer algorithm to calculate x^n (n is a positive integer) in $O(\log n)$ time. You may assume that multiplying any two numbers can be done in $\Theta(1)$ time.

Problem 7 (10 points).

Given two *sorted* arrays A and B of size m and n respectively, design an algorithm to find the median of A and B. Your algorithm should run in $O(\log(m+n))$ time.

Problem 8 (10 points).

Given an array of points $P = [p_1, p_2, \cdots, p_n]$ on 2D plane, where point $p_i = (x_i, y_i)$ is given as its x- and y-coordinates, and integer k, $1 \leq k \leq n$, design an algorithm to find the closest k points to the origin in P. Your algorithm should run in $O(n)$ time.