1. (8 points) Consider sorting $n$ numbers stored in array $A$ by first finding the smallest element of $A$ and exchanging it with the number stored in $A[1]$. Then find the second smallest element of $A$, and exchange it with $A[2]$. Continue in this manner for the first $n - 1$ elements of $A$. This algorithm, called **Selection-Sort**, is given below. Assume the input and output are as specified for the sorting problem on page 16 of our textbook.

**Selection-Sort**($A$):
1. $n = A$.length
2. for $j = 1$ to $n - 1$
3. smallest = $j$
4. for $i = j + 1$ to $n$
5. if $A[i] < A[\text{smallest}]$
6. smallest = $i$
7. exchange $A[j]$ with $A[\text{smallest}]$

(a) (1 point) Give the line number(s) of the basic operation in the algorithm

(b) (2 points) Write a nested summation to describe the number of times the basic operation of **Selection-Sort** is executed on an input of size $n$ and then express this summation as a closed-form solution.

(c) (2 points) Why does the outer for loop of the algorithm need to run for only the first $n - 1$ elements?

(d) (3 points) Are there best-case and worst-case asymptotic running times for **Selection-Sort**? If so, give the best-case and worst-case running times of the algorithm in $\Theta$ notation. If not, give the worst-case running time in $\Theta$ notation. Explain your answer.
2. (4 points) Answer true or false. Use the definitions of $O$, $\Theta$, and $\Omega$ to determine whether the following assertions are true or false. Briefly justify your answers.

a. $\frac{n^2(n+1)}{2} \in O(n^3)$

b. $\frac{n(n+1)}{2} \in O(n)$

c. $\frac{n(n+1)}{2} \in \Theta(n^3)$

d. $\frac{n(n+1)}{2} \in \Omega(n)$

3. (4 points) For each of the following functions, indicate the set $\Theta(g(n))$ the function belongs to. Show your work in simplifying these expressions to arrive at each answer.

a. $(n^2 + 1)^{10}$

b. $\sqrt{10n^2 + 7n + 3}$

c. $2^{n+1} + 3^{n-1}$

d. $[\lg n]$