1. (10 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.

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

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

(b) Are there best-case and worst-case running times for Selection-Sort? Answer “yes” or “no” and explain your answer.

(c) If you answered “yes” to part (b), give the best-case and worst-case running times of the algorithm in $O$ and $Ω$ notation. If you answered “no” to part (b) give the running time for Selection-Sort in $Θ$ notation.
2. (5 points) Rank the functions given below by decreasing order of growth; that is, find an arrangement $g_1, g_2, g_3, g_4$ of the functions satisfying $g_1 = \Omega(g_2) = \Omega(g_3) = \Omega(g_4)$.

$$2^n \quad n^{\frac{3}{2}} \quad n \lg n \quad n^\lg n$$

In the space below, list the functions given above in terms of decreasing running time (highest to lowest, left to right), as $n$ increases to $\infty$ (justify your answers):