

CS375 Assignment 2

Due Sep 21st 10PM ET

A general note for CS375: When writing up your homework, please write neatly and explain your answers clearly, giving all details needed to make your answers easy to understand. Graders may not award full credit to incomplete or illegible solutions.

You are encouraged to discuss the problems with your classmates, **but your submission should be your own**. When writing your solutions up, you should not be looking at any other student's submission.

Clear communication **is the point**, on every assignment. In general in CS375, unless explicitly specified otherwise, answers should be accompanied by explanations. Answers without explanations may not receive full credit. Please feel free to ask me any questions about explanations that might come up!

Necessary acknowledgements: many of these questions are reproductions or adaptations of questions of Eric Aaron's devising. Others are from our course texts as listed on the syllabus.

1. Use induction to argue that the following algorithm correctly sorts a list:

```
SelectionSort(A[0:n]):  
    for i in range(n):  
        min_index = i  
        for j in range(i + 1, n):  
            if A[j] < A[min_index]:  
                min_index = j  
        if min_index != i:  
            swap A[i] with A[min_index]
```

2. Additionally, determine `SelectionSort`'s runtime. Note that I'm leaving out the word best-case and worst-case here because its runtime as a function of the input size is independent of the actual input.

3. Suppose we know that algorithm A has worst-case runtime of $O(n^2)$ and a best-case runtime of $O(n \log n)$.

(a) Could A 's best-case runtime be $\Omega(n^2)$? Explain your answer.

(b) Could A 's best-case runtime be $\Theta(n)$? Explain your answer.

(c) Could A 's worst-case runtime be $\Theta(n^3)$? Explain your answer.

(d) Could A 's worst-case runtime be $\Theta(1)$? Explain your answer.