Sabancı University
Faculty of Engineering and Natural Sciences

CS301 – Algorithms

Homework 2

Due: March 12, 2024 @ 23.55
(upload to SUCourse)

PLEASE NOTE:

- Provide only the requested information and nothing more. Unreadable, unintelligible, and irrelevant answers will not be considered.
- Submit only a PDF file. (-20 pts penalty for any other format)
- Not every question of this homework will be graded. We will announce the question(s) that will be graded after the submission.
- You can collaborate with your `TA/INSTRUCTOR ONLY` and discuss the solutions of the problems. However, you have to write down the solutions on your own.
- Plagiarism will not be tolerated.

Late Submission Policy:

- Your homework grade will be decided by multiplying what you normally get from your answers by a "submission time factor (STF)".
- If you submit on time (i.e. before the deadline), your STF is 1. So, you don't lose anything.
- If you submit late, you will lose 0.01 of your STF for every 5 mins of delay.
- We will not accept any homework later than 500 mins after the deadline.
- SUCourse's timestamp will be used for STF computation.
- If you submit multiple times, the last submission time will be used.

Question 1

(a) What is the form of the input array that triggers the worst case of the insertion sort?

(b) What is the complexity of this worst–case behavior in Θ notation?

(c) Explain how this particular form of the array results in this complexity.

Question 2

(a) What is the form of the input array that triggers the best case of the insertion sort?

please proceed to the next page $\longrightarrow$

(b) What is the complexity of this best–case behavior in Θ notation?

(c) Explain how this particular form of the array results in this complexity.

Question 3

Which of the following sorting algorithms are stable: insertion sort, merge sort, heapsort, and quicksort? Give a simple scheme that makes any comparison sort stable. How much additional time and space does your scheme entail?

Question 4

(a) Given n d-digit numbers in which each digit can take on up to k possible values, RADIX-SORT correctly sorts these numbers in .... time if the intermediate stable sorting algorithm is Counting Sort.

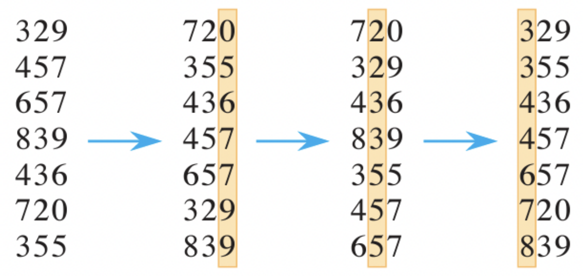

Figure 1: The operation of radix sort on seven 3-digit numbers. The leftmost column is the input. The remaining columns show the numbers after successive sorts on increasingly significant digit positions. Tan shading indicates the digit position sorted on to produce each list from the previous one.

(b) Using Figure 1 as a model, illustrate the operation of RADIX-SORT on the following list of English words: COW, DOG,SEA, RUG, ROW, MOB, BOX, TAB, BAR, EAR, TAR, DIG, BIG, TEA, NOW, FOX.

Question 5

The pseudo-code for Quicksort algorithm is given below.

Algorithm 1 Quicksort algorithm

```
Function Quicksort(array A, l, r):
    if $r - l + 1 \leq 1$ then
        return
    end
    $p \leftarrow$ ChoosePivot($A$, $l$, $r$)
    Partition($A$, $p$, $l$, $r$)
    Quicksort($A$, $l$, $p - 1$)
    Quicksort($A$, $p + 1$, $r$)

Function Partition($A$, $p$, $l$, $r$):
    $i \leftarrow l + 1$
    for $j \leftarrow l + 1$ to $r$ do
        if $A[j] \leq p$ then
            swap $A[i]$ with $A[j]$
            $i \leftarrow i + 1$
        end
    end
    swap $A[i - 1]$ with $p$
    return $i - 1$
Function ChoosePivot($A$, $l$, $r$):
    return $A[\lfloor (l + r)/2 \rfloor]$
```

(a) Write down the recurrence for the running time for the case where the algorithm chooses the median as the pivot at each iteration.

please proceed to the next page ⟶

(b) Calculate a tight bound for this recurrence using the Master Theorem.

(c) [5 points] Write down the recurrence for the running time for the case where the algorithm chooses the smallest element in the array as the pivot at each iteration.

(d) [5 points] Calculate a tight bound for this recurrence using the iteration method.