

Due 16th June 2022 at 4pm Sydney time

In this assignment we review some basic algorithms and data structures, and we apply the divide-and-conquer paradigm. There are *four problems* each worth 20 marks, for a total of 80 marks.

Your solutions must be typed, machine readable PDF files. *All submissions will be checked for plagiarism!*

For each question requiring you to design an algorithm, you *must* justify the correctness of your algorithm. If a time bound is specified in the question, you also *must* argue that your algorithm meets this time bound.

Partial credit will be awarded for progress towards a solution.

Question 1

Your friend attempted to send you an array of n bits, starting with a 0, and alternating between 0s and 1s. However, due to the network being unreliable, one of the bits was not sent. You received the array A of length $n - 1$, and you would like to determine which bit was not sent.

1.1 [2 marks] Suppose $n = 8$, and you received the array $A = [0, 1, 0, 1, 1, 0, 1]$. Identify the bit that was not sent by its 1-based index in the array your friend tried to send you.

1.2 [5 marks] Suppose $n = 20$, and you received an array in which $A[10] = 0$. Which of the twenty bits could have been omitted from the original array? Provide reasoning to justify your answer.

1.3 [13 marks] Design an algorithm which runs in $O(\log n)$ time and finds the index of the missing bit.

Question 2

You are given an array A of n integers. You are required to find indices i, j, k (not necessarily distinct) such that $A[i] + A[j] = A[k]$, or return that no such indices exist.

Design algorithms which solve this problem and run in:

2.1 [6 marks] worst case $\Theta(n^2 \log n)$ time.

2.2 [6 marks] *expected* $\Theta(n^2)$ time.

2.3 [8 marks] worst case $\Theta(n^2)$ time.

Question 3

You are given an array A containing each integer from 1 to n exactly once. Your task is to compute $f(A)$, the sum of $\max(A[\ell..r])$ over all pairs of indices (ℓ, r) such that $\ell \leq r$.

3.1 [2 marks] Suppose $n = 3$ and the array is $A = [2, 1, 3]$. Determine the value of $f(A)$.

For 3.2 and 3.3, suppose i and j are indices such that $1 \leq i \leq j \leq n$, and let $g(i, j)$ be the number of subarrays $A[\ell..r]$ where $r > j$ and the maximum value is $A[i]$.

3.2 [4 marks] For a given pair of indices (i, j) , under what conditions is $g(i, j)$ nonzero? In other words, what is the criterion for $A[i]$ to be the maximum of some subarray with its right endpoint at an index greater than j ?

3.3 [6 marks] Given an index j , design an algorithm which runs in $O(n)$ time and determines the values of $g(i, j)$ for all $i < j$.

3.4 [8 marks] Design an algorithm which runs in $O(n \log n)$ time and determines the value of $f(A)$.

Question 4

Your friend has constructed an array A of n distinct integers, where $n \geq 2$. However, you cannot access the elements of the array directly; instead, they instead only allow you to ask questions of the form “What is the maximum value amongst $A[l], A[l + 1], \dots, A[r - 1], A[r]$?”, where you may choose any valid indices l and r such that $l \leq r$. You may assume any questions you ask are answered in constant time.

Your goal is to determine the value of the **second largest** element in the array.

4.1 [2 marks] How can you find the value of the largest element in the array using only one question?

4.2 [2 marks] If you know that the largest element occurs at index i , how could you then find the value of the second largest element using only two questions?

4.3 [11 marks] Design an algorithm which runs in $O(\log n)$ time and determines the value of the second largest element in the array.

4.4 [5 marks] Now your friend imposes an extra restriction: for each question you ask except the first, the value of $r - l$ should be no larger than the value of $r - l$ in the previous question. Subject to this restriction, design an algorithm which runs in $O(\log n)$ time and determines the value of the second largest element in the array.

You may choose to skip 4.3, in which case we will mark your submission for 4.4 as if it was submitted for 4.3 also.