M. Tech. Mid Term. 90 mins. 40 points.

(1) There is a D&C algorithm for finding x^n, $n \geq 1$, in integer.
Find x^n and x^{2n} and compute x^n as $x^{2n} \cdot x^{n-1}$.

(a) Write a Recursive procedure Exp(x, n) implementing the idea above. (4)

(b) Let $T(n)$ be the number of multiplications done in Exp(x, n). Write down the recurrence for $T(n)$ and solve it exactly for $n = 2^k$.

(c) Extend the result of (b) and solve exactly for all n by explicit algebra or induction. (TF) (3)

(2) Let $A[1..n]$ be an array of n integers. An value i is a valley index if $A[i-1] > A[i] > A[i+1]$. Consider time the left and the right neighbors when they exist. Design an $O(\log n)$ divide and conquer algorithm to locate a valley index in A. Present the algorithm in a pseudocode and prove the correctness. (8+2)

(3) Design an incremental algorithm that finds Maximum and Minimum of an array $A[1..n]$ of distinct integers, same pseudocode. The complexity of your algorithm must be $\frac{3n}{2} - 2$ for odd even n. What is the complexity for odd n? (6+2+2)

(4) (a) Let $f(n)$ and $g(n) \in O(L(n))$
Is $f(n) - g(n) \in O(L(n))$?
Prove your claim.

(b) Let $f(n)$ and $g(n) \in \Theta(L(n))$
Is $f(n) - g(n) \in \Theta(L(n))$?
Prove your claim.

(c) Prove that $H(n) = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$ is $O(\log n)$.

(d) Expand and Simplify
$T(n) = 2\,T(\frac{n}{2}) + n/\log n$
for $n = 2^k$ and obtain closed form formula for $T(n)$.
(1+2+7+6)

Exp(x, n)
if n = 0 ; 1 ret.
if (n = 1)
return (x)
else
$n_1 = \lfloor n/2 \rfloor$; $n_2 = \lceil n/2 \rceil$
a = Exp(x, n_1)
b = Exp(x, n_2)
return (a·b)

(2 marks. must be a clean recursive call. Give one base case and recursive cases are important. They must show two recursive calls and multiply the returned values). (No partial Marks)

b) $T(n) = T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 1$, if $n > 1$
$= 0$ if $n = 1$. (No partial Marks)

c) When n is a power of 2, say 2^k,

$$\begin{aligned} T(2^k) &= 2\,T(2^{k-1}) + 1 \\ &= 2\left[2\,T(2^{k-2}) + 1\right] + 1 \\ &= 2^2\,T(2^{k-2}) + 2 + 1 \\ &= 2^3\left[2\,T(2^{k-3}) + 1\right] + 2 + 1 \\ &= 2^3\,T(2^{k-3}) + 2^2 + 2 + 1 \\ &\vdots \\ &= 2^k\,T(2^{k-k}) + 2^{k-1} + 2^{k-2} + \cdots + 2 + 1 \\ &= 0 + 2^{k-1} + 2^{k-2} + \cdots + 2 + 1 \\ &= 2^k - 1. \\ &= n - 1 \end{aligned}$$

(This derivation will be explicit comp. must be shown. Ans is exactly $n-1$. No answers like $O(n)$ etc are acceptable. If want exact solution. No partial Marks.

$T(n) = n - 1$ for $n \geq 1$.

Test for $n = 1$. (Base for Induction). (Must be stated).
Assume $T(k) = k - 1$ for all $1 \leq k \leq n-1$. (Strong induction Hypothesis).

$$\begin{aligned} T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 1 \\ &= \lfloor n/2 \rfloor - 1 + \lceil \tfrac{n}{2} \rceil - 1 + 1 \quad (\text{by IH}) \\ &= \lfloor n/2 \rfloor + \lceil n/2 \rceil - 1 \\ &= n - 1. \end{aligned}$$

---

(1) $A[n-1] < A[n] < A[n+1]$
(2) $A[n-1] < A[n] > A[n+1]$
(3) $A[n-1] > A[n] < A[n+1]$
(4) $A[n-1] > A[n] > A[n+1]$

In case 1, they must recurse on $A[n..n]$. (because $n+1$ may be a valley index).

In case 2, they may proceed in any one half. (because $n+1$ may be a valley index).

In case 2, they may proceed in any one half.
In case 3 — the most pattern in end half.

In base case
— $n = 1$ — return 1.
— $n = 2$ — return min index. $A[1] \geq A[2]$ return 1 else return 2.
— $n \geq 3$ Case 1 — return 1
case 2 — $1 \cdot n/2$.
case 3 — return 2.
Case 4 — return 3.

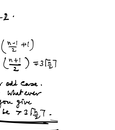
When $n \geq 4$
$m = \lfloor n/2 \rfloor$.
and Proceed with $A[m..n]$ or with $A[1..m]$
by return m as discussed above.

Base cases must be explicitly stated for 1, 2 and 3.

Recursion must be properly called.

Proof of correctness is important explicit argument.

For example: In case 1, $A(m)$, $A(m)$, $A(m+1)$ form a decreasing sequence. If it decreases all the way to the last $A[1]$ is valley index. If it increases, may above, but still we find valley point. Thus a valley index exists in the left and we can proceed in recursively with $A[1..m]$.



Similar argument must be given for all cases. No partial Marks.

Very logical error or incomplete are zero.

---

(3) Minmax (A[1..n], n).
if (n = 1)
min = max = A[1]
if n = 2
if A[1] > A[2]
min = A[2]
max = A[1]
else
min = A[1]
Max = A[2].
for i = 2 to $\lfloor n/2 \rfloor$
if A[2i-1] > A[2i]
if A[2i-1] > max
max = A[2i-1]
if A[2i] < min
min = A[2i]
else
if A[2i] > max
Max = A[2i]
if A[2i-1] < min
min = A[2i-1].
if (n is odd).
if A[n] > max
max = A[n]
if A[n] < min
min = A[n].

This is one of the recommended Code.
Other correct Codes may be accepted.
But the complexity is $> \frac{3n}{2} - 1$, or $> \frac{3n}{2}$.

Zero

Complexity for n even:
$1 + 3\left(\frac{n}{2} - 1\right) = \frac{3n}{2} - 2$.

Complexity for n odd
$1 + 3\left(\frac{n-1}{2}\right) + 2 = 3\left(\frac{n-1}{2} + 1\right)$
$= 3\left(\frac{n+1}{2}\right) = 3\lceil \frac{n}{2} \rceil$

It is possible to reduce even for odd cases. (i.e. Minmax Pairs) — we ask what exactly the complexity of odd, you give. Wrong. It should not be $> 3\lceil \frac{n}{2} \rceil$.

---

(4) (a) Yes
$f(n) \leq c_1\,L(n)$
$g(n) \leq c_2\,L(n)$
$f(n) - g(n) \leq f(n) + g(n)$
$\leq (c_1 + c_2)\,L(n)$ || Always true. Proof Easy.

(They MUST NOT write
$f(n) - g(n) \geq (c_1 - c_2)\,L(n)$
This is wrong.
The inequality is logically incorrect. If this is done Zero

(b) No
ex. $f(n) = n^2 + 1 \quad - O(n^2)$
$g(n) = n^2 \quad - O(n^2)$.
But $f(n) - g(n) = 1$ is $O(n^0)$
You consider example to be.

(c) Any proof is ok.

(d) $T(n) = 2\,T(\frac{n}{2}) + n/\log n$

$$\begin{aligned} &= 2\left[2\,T\left(\tfrac{n}{4}\right) + \tfrac{n/2}{\log \frac{n}{2}}\right] + \tfrac{n}{\log n} \\ &= 2^2\,T\left(\tfrac{n}{2^2}\right) + \tfrac{n}{\log \frac{n}{2}} + \tfrac{n}{\log n} \\ &= 2^3\,T\left(\tfrac{n}{2^3}\right) + \tfrac{n}{\log \frac{n}{4}} + \tfrac{n}{\log \frac{n}{2}} + \tfrac{n}{\log n} \\ &\vdots \\ &= 2^k\,T(1) + \tfrac{n}{1} + \tfrac{n}{2} + \cdots + \tfrac{n}{\log n} \\ &= n \cdot c + n\left(\tfrac{1}{1} + \tfrac{1}{2} + \cdots + \tfrac{1}{\log n}\right) \\ &= n \cdot c + n \cdot O(\log \log n) \\ &= O(n \log \log n). \end{aligned}$$

Here we use $H(n) = 1 + \frac{1}{2} + \cdots + \frac{1}{n} = O(\log n)$

So $1 + \frac{1}{2} + \cdots + \frac{1}{\log n} = O(\log \log n)$

No partial Marks.