ZIO 2023 Editorial

Problem 1

If a person gets m questions correct, their final marks lies in the range $[m \cdot X - (N - X), m \cdot X]$

The problem is to compute the size of the union of all such ranges. This can be done simply manually, but a formula exists as well.

For test case 1, the ranges are
$[-7, 0], [-2, 4], [3, 8], [8, 12], [13, 16], [18, 20], [23, 24], [28, 28]$

The union of these can be simply calculated to get the answer.

For test case 2, you can note that the ranges will always be disjoint (because $m \cdot X - (N - X)$ is guaranteed to be larger than $(m - 1) \cdot X$), so the answer is merely the sum of the ranges.

For test case 3, you can proceed similarly to test case 1, or you can try to derive a formula which works as
$N + 1 + min(N, X) + min(N - 1, X) + \ . \ . \ . + min(1, X)$

For $X \leq N$, this becomes equivalent to
$(N + 1) + 1 + 2 + ... + X + X \cdot (N - X) = (N + 1) + X(X + 1)/2 + X(N - X)$

For $X > N$, it simply becomes $\frac{(N+1)(N+2)}{2}$.

Problem 2

Suppose we fixed the k distinct badge numbers to be $a_1, a_2, \ldots a_k$, and the frequency of the people with those badge numbers to be $f_1, f_2, \ldots f_k$. Then the contribution to the sum for this set of k badge numbers is $(2^{f_1} - 1)(2^{f_2} - 1)\ldots(2^{f_k} - 1)$ (for every badge, there are 2^f ways to choose a subset of the people and $-\ 1$ for the empty subset since there must be atleast one present)

For test case 1, you can simply just bruteforce all possible badges with the $2^f - 1$ formula

For test case 2, note that all f_i are equal, thus implying that the contribution to sum for a fixed set of badges will always be equal to $(2^4 - 1)^3$.

Thus, we only need to multiply this contribution to sum by the number of ways we can choose a fixed set of badges, that is $C(5, 3)$

The final test case can be solved with some casework on how many badges we take with a certain frequency however the intended solution is $dp[i][j]$ = number of ways to take exactly j distinct badges from first i badges.

$$dp[i][j] \ = \ dp[i-1][j] \ + \ dp[i-1][j-1] \cdot (2^{f_i} - 1)$$

Problem 3

Let the minimum of the array be at index m. Observe that for any l and r such that $1 \leq l \leq m$ and $m \leq r \leq n$, the index of the minimum element over $[l, r]$ will be m.

Further, the minimum indices of subarrays within $[1, m - 1]$ and $[m + 1, N]$ are only dependent on the relative order of elements within these subarrays and not the actual elements. There are $\binom{N-1}{m-1}$ different ways of choosing the $m - 1$ numbers that will appear in indices $[1, m - 1]$ (and equivalently, also the remaining indices that will appear in $[m + 1, N]$).

Once we have chosen the way the numbers are partitioned between left and right, we can recursively keep dividing the left and rights sides into a minimum element, their own left side and right side, to compute the number of orderings.

Finally, to compute the answer we can multiply all the ways of making choices over all partitions.

(a) $N = 4$

l	r	m (index of minimum element)	Number of partitions
1	4	3	$\binom{3}{2} = 3$
1	2	2	$\binom{1}{1} = 1$

The answer is $3 \cdot 1 = 3$.

(b) $N = 8$

l	r	m (index of minimum element)	Number of partitions
1	8	5	$\binom{7}{4} = 35$
1	4	1	$\binom{3}{0} = 1$
2	4	2	$\binom{2}{0} = 1$
3	4	3	$\binom{1}{0} = 1$
6	8	7	$\binom{2}{1} = 2$

The answer is $35 \cdot 1 \cdot 1 \cdot 1 \cdot 2 = 70$.

(c) $N = 15$

l	r	m (index of minimum element)	Number of partitions
1	15	14	$\binom{14}{13} = 14$
1	13	4	$\binom{12}{3} = 220$
1	3	2	$\binom{2}{1} = 2$
5	13	10	$\binom{8}{5} = 56$
5	9	9	$\binom{4}{4} = 1$
5	8	8	$\binom{3}{3} = 1$
5	7	7	$\binom{2}{2} = 1$
5	6	5	$\binom{1}{1} = 1$
11	13	13	$\binom{2}{2} = 1$
11	12	11	$\binom{1}{1} = 1$

The answer is $14 \cdot 220 \cdot 2 \cdot 56 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 344960$.

Problem 4

The relationship "Stone A_i weighs X_i units more than Stone B_i" can be represented as an edge in a graph, where the nodes are the stones.

Within each connected component of this graph, if the weight of any one stone is fixed the weights of all other stones will be uniquely determined. Specifically, consider a connected components of nodes $\{u_1, u_2, \ldots u_k\}$. Each stone u_i will have some relative weight w_i, and we can simultaneously shift the weight of all stones by any number W, so that the final weight of each stone u_i is $W + w_i$.

We want to minimize the sum of weights of the stones, so within each component it is optimal for the lightest stone to have weight 1. So, the value of W should be equal to $1 - \min_{i=1..k} \{w_i\}$.

Each component of the graph will have its own value of W. The final weight of each node x should be computed based on the relative weight of node x, and the W value of the component in which x exists. We can add up the resultant weights of all the nodes to yield the answer for one step.

When a step is performed, two components may be merged. So, the relative weights of elements in one component should be adjusted to be consistent with the other component, and the total contribution of only these two components might change. We can adjust the total weight of the previous step to yield the total weight of the current step by adding the new contribution of the combined component, and subtracting the individual contributions of the old components.