Suppose that $g = \gcd(a_1, a_2, ..., a_n)$, the answer we seek is the minimum value of

$$g \cdot \left(\frac{a_1}{g} + \frac{\gcd(a_1, a_2)}{g}\right) + ... + g \cdot \frac{\gcd(a_1, a_2, ..., a_n)}{g}.$$

Suppose we let $a_i' = \frac{a_i}{g}$ for each $1 \leq i \leq n$, then this value is equal to

$$g \cdot (a_1' + \gcd(a_1', a_2') + ... + \gcd(a_1', a_2', ..., a_n')).$$

Note that now we have $\gcd(a_1', a_2', ..., a_n') = 1$.

Now, consider the following greedy algorithm. We will start with an initially empty array b and add to the end of array b the element that minimizes the GCD with the already existing array b . It can be observed that the gcd will reach 1 in at most 10 iterations. After that, the remaining elements can be added in any order.

Claim 1: At most 17 rounds are needed.

Proof It actually suffices to show that after appending each element to the back of b, the GCD decreases. Then, we know that it will decrease at most $\lfloor \log_2(100000) \rfloor = 16$ times, which a gives a maximum of 17 times in total.

This can be shown using proof by contradiction. Observe that the first element appended into b is the minimum value among all a_i's. Thus, if after a round after the first, the gcd of all values in b doesn't decrease and remains unchanged (note that it is decreasing, though not strictly), then there must be an integer $d > 1$ that divides all values a_i. This contradicts with our assumption that $\gcd(a_1', a_2', ..., a_n') = 1$.

Hence, the claim is proved. □

Claim 2: The greedy algorithm gives one of the optimal solutions.

Proof Consider a specific permutation $(b_1, b_2, ..., b_n)$ of $(a_1', a_2', ..., a_n')$. Suppose this gives an answer of A. Now, if there is an index $1 \leq i < n$ such that $\gcd(b_1, b_2, ..., b_{i-1}, b_i) > \gcd(b_1, b_2, ..., b_{i-1}, b_{i+1})$, then observe that if we exchange the positions of b_i and b_{i+1}, only the part $\gcd(b_1, b_2, ..., b_{i-1}, b_i)$ that makes up the original answer A changes. In this case, it actually decreases.

Thus, as long as there is such an index i, we can keep performing the swapping operation until no more such i exists. By this time, it is clear that the answer is minimised.

This means the method adopted in the greedy solution is correct. □