- Always try to give algorithm with best possible running time. The points that you obtain will depend on the running time of your algorithm. For example, a student who gives an $O(n)$ algorithm will receive more points than a student who gives an $O(n^2)$ algorithm.

- **Use of unfair means will be severely penalized.**

There are 3 questions for a total of 50 points.

(15) 1. Given integers $a_1, ..., a_n$, design an algorithm that determines whether there exists a partition of the set $\{1, ..., n\}$ into 3 disjoint subsets P, Q, and R such that

$$\sum_{i \in P} a_i = \sum_{j \in Q} a_j = \sum_{k \in R} a_k.$$

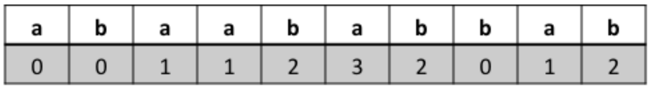
Discuss running time of your algorithm.

(15) 2. Given positive integers n and W, design an algorithm that finds the number of n element sets $\{x_1, ..., x_n\}$ such that $\sum_i x_i^2 = W$ and all x_i's are integers. Discuss running time of your algorithm.

(20) 3. You are given a string S of alphabets containing n alphabets. Let $S[i]$ denote the i^{th} alphabet of the string S and let $S[1, i]$ denote the substring $S[1]...S[i]$. Design an algorithm that computes the following:

For each i, the length of the longest *proper* prefix of $S[1, i]$ that is also a suffix of $S[1, i]$.

Discuss running time of your algorithm.



a	b	a	a	b	a	b	b	a	b
0	0	1	1	2	3	2	0	1	2

Figure 1: Example string with 10 alphabets. The integer below the i^{th} alphabet denotes the length of the longest proper prefix of $S[1, i]$ that is also a suffix of $S[1, i]$.