Problem 5. «A simple hash function»

Carol invented a new hash function. The key $k = (k_1, \dots, k_6)$ for this hash function is a binary vector of length 6. The input for the hash function is a sequence of digits. It should be divided into blocks of length 6. If the length of the sequence is not a multiple of 6 then it can be completed with 1, 2, 3, and so on up to the necessary length. For example, if an input is 7256 then it should be changed to 725612 first.

Then each block of input, say $(p_1, \dots, p_6)$ should be transformed into the number by the rule: $(-1)^{k_1} \cdot p_1 + \dots + (-1)^{k_6} \cdot p_6$. Here $(-1)^0 = 1$ and $(-1)^1 = -1$. Results of such calculations for blocks, say $n_1, n_2, n_3, n_4, \dots$, then form a resulting hash value as $H = n_1 - n_2 + n_3 - n_4 + \dots$.

For example, if the key is (001101) then hash for the sequence 134875 512293 is $H = (1 + 3 - 4 - 8 + 7 - 5) - (5 + 1 - 2 - 2 + 9 - 3) = -6 + 8 = 2$.

Carol applied her hash function in the system for logging at the bank website. Every user enters his password, say P, then system counts hash value $H(P, K)$ and if it coincides with the hash value from the data base, then user is logged.

But after some time Carol realized that the system is not secure. Malefactors can construct collisions and enter the system illegally. How do they do it? Propose a simplest algorithm how to get a collision of the first order for any known input sequence P if the key K is unknown.

By the way, find the shortest collision for the sequence from the example, $P = 134875\,512293$. Since K is unknown, the hash value $H(P, K)$ is still unknown to you.

The picture is taken from https://www.lepuchin.com/Security-Brief-P13-Birthday-Attack.