

☆ Whole Minute Dilemma

Your music player allows you to choose songs to play, but only in pairs and only pairs of songs with durations that add up to whole minutes. Given a list of song durations (in seconds), you must calculate the total number of different pairs of songs that can be chosen.

For example, given song lengths [40s, 20s, 60s], there is one pair of songs that can be chosen [40s, 20s]. While the third song is a whole minute long, songs **must** be chosen in pairs.

Function Description

Complete the function *playlist* in the editor below. The function must return a single integer which is the number of ways to choose two songs such that the total duration is a multiple of a whole minute.

playlist has the following parameter(s):

songs[songs[0],...songs[n-1]]: an array of integers wherein each element denotes duration of a song in seconds

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq \text{songs}[i] \leq 1000$, where $0 \leq i < n$

► Input Format For Custom Testing

▼ Sample Case 0

Sample Input For Custom Testing

```
4
10
50
90
30
```

Sample Output

```
2
```

Explanation

The first and second songs pair to 60 seconds. The third and fourth songs pair to 120 seconds. No other pairs will satisfy the requirement.

► Sample Case 1

► Sample Case 2

YOUR ANSWER

Draft saved 03:19 pm

Original code

Java 7



```
1  ▶ import ↔;
8
9  public class Solution {
10
11
12  |    // Complete the playlist function below.
13  ▼    static long playlist(List<Integer> songs) {
14
15
16      }
17
18
19  ▶    public static void main(String[] args) throws IOException {↔}
40  }
41
```