You have N (0 ≤ N ≤ 10) wooden sticks, which are labeled from 1 to N. The i-th stick has a length of Li (0 ≤ Li ≤ 1,000,000). Your friend has challenged you to a simple game: you will pick three sticks at random, and if your friend can form a triangle with them (degenerate triangles included), he wins; otherwise, you win. You are not sure if your friend is trying to trick you, so you would like to determine your chances of winning by computing the number of ways you could choose three sticks (regardless of order) such that it is impossible to form a triangle with them.

* **[time limit] 3000ms (cs)**
* **[input] integer n**

Number of sticks. N can be 0.

* **[input] array.integer L**

Integers L1, ..., LN

* **[output] integer**

Output a single line containing the number of triples.

<https://codefights.com/challenge/pGKGLqeAEgxgeQG9H/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int Triangle(int n, int[] L)

{

int ans = 0;

for (int i = 0; i < n - 2; i++)

{

for (int j = i + 1; j < n - 1; j++)

{

for (int k = j + 1; k < n; k++)

{

//propiedad de la desigualdad triangular

int a = L[i], b = L[j], c = L[k];

bool cond = a + b > c && b + c > a && c + a > b;

if (cond)

{

ans++;

}

}

}

}

return ans;

}

static void Main(string[] args)

{

int n = 4;

int[] L = { 5, 2, 9, 6 };

Console.WriteLine(Triangle(n, L));

Console.ReadLine();

}

}

}