**pseudocode**

**my\_len(A) T(n) =1 + n \* 1 + 1 = n+2**

//Array A is passed

count ← 0 1

for each element in A n

count ← count + 1 1

return count 1

**Algorithm is\_triangular(A) T(n) = n + 2 + n \* n \* n = n3 + n + 2**

//Array A is passed

n ← my\_len(A) n + 2 = n + 2

for i ← 0 to n-1 n

for j ← i+1 to n-1 n

for k ← j+1 to n-1 n

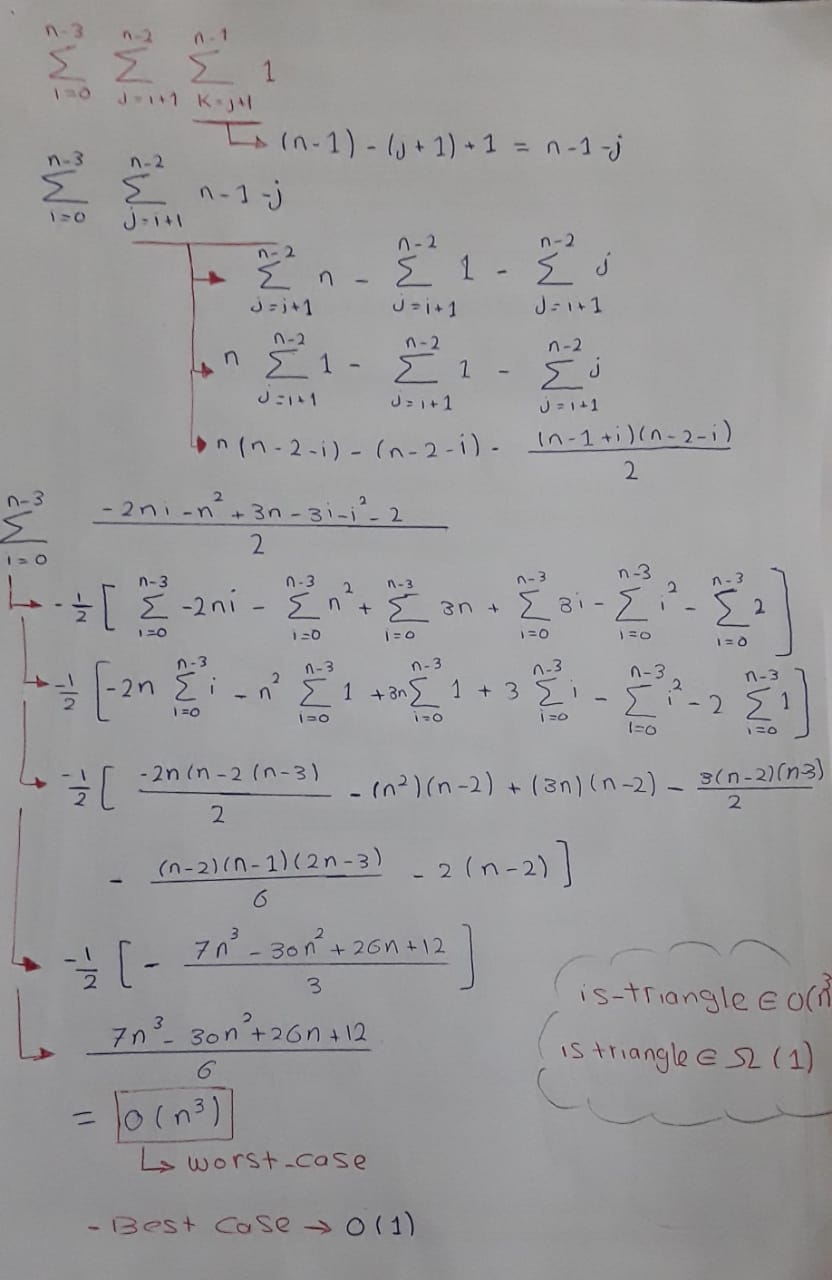
if A[i] + A[j] > A[k] and A[i] + A[k] > A[j] and A[j] + A[k] > A[i]

return 1 1

return 0

**Time complexity = T(n) =n3 +n + 2**

**Analysis**



**Best Case** **A screenshot of a computer program

Description automatically generated**

**Worst Case**

A screenshot of a notebook

Description automatically generated