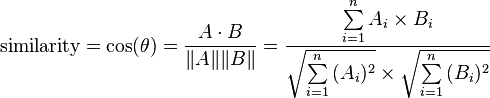
445. Cosine Similarity

* [Description](http://lintcode.com/en/problem/cosine-similarity/" \l "description)
* [Notes](http://lintcode.com/en/problem/cosine-similarity/#note)
* [Testcase](http://lintcode.com/en/problem/cosine-similarity/#testcase)
* [Judge](http://lintcode.com/en/problem/cosine-similarity/#judge)

Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0° is 1, and it is less than 1 for any other angle.

See wiki: [Cosine Similarity](https://en.wikipedia.org/wiki/Cosine_similarity)

Here is the formula:



Given two vectors A and B with the same size, calculate the cosine similarity.

Return 2.0000 if cosine similarity is invalid (for example A = [0] and B = [0]).

Have you met this question in a real interview?

Yes

**Example**

Given A = [1, 2, 3], B = [2, 3 ,4].

Return 0.9926.

Given A = [0], B = [0].

Return 2.0000

<http://lintcode.com/en/problem/cosine-similarity/#>

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication1;

import java.util.\*;

public class JavaApplication1 {

public static double cosineSimilarity(int[] A, int[] B) {

// write your code here

if(A.length ==0 && B.length ==0) return 2.0000;

boolean flag = true;

for(int i =0; i<A.length; i++) {

if(A[i] != 0 || B[i] !=0) {

flag = false;

}

}

if(flag) return 2.0000;

double numerador =0;

double s1=0, s2 =0;

for(int i =0; i<A.length; i++) {

numerador += (A[i] \* B[i]);

s1 += (A[i] \* A[i]);

s2 += (B[i] \* B[i]);

}

s1 = Math.sqrt(s1);

s2 = Math.sqrt(s2);

double res = numerador/(s1 \* s2);

return (double)Math.round(res \* 10000d) / 10000d;

// return Double.parseDouble(String.format("%.2f", numerador/(s1\*s2)));

}

public static void main(String[] args) {

// TODO code application logic here

//key="abcd" and size=100,

//int[] A = {1, 2, 3};

//int[] B = {2, 3 ,4};

int[] A ={};

int[] B ={};

double res = cosineSimilarity( A, B);

System.out.println(res);

}

}