

In the previous documents, I used ChatGpt to create some tests for my Distance3d function. In the same chat thread, I decided to give it some examples of “good” documentation and asked it to write me a .md readme file for the function:

NE

Below are some examples of good documentation readme.md files. Write a documentation .md file for the Distance3d function described above, making sure to contain the following four sections: Syntax, Description, Examples, Theory.

As an example, I gave it my handwritten documentation for KE.md and GPE.md



Here is an example of a documentation `.md` file for the `Distance3d` function:



`Distance3d`

Calculates the distance between two points in a 3d space.

Syntax

Copy code

```
[r] = Distance3d(A,B)
```

Description

`A` and `B` are 3d points represented by `[x,y,z]`.

Examples

To find the distance between points `A` and `B`:

Copy code

```
A = [1, 2, 3];  
B = [4, 5, 6];  
distance = Distance3d(A, B);
```

The distance between points `A` and `B` is `5.830951894845301`.

Theory




The distance between two points in 3d space is calculated using the Pythagorean theorem:

Copy code

```
d = sqrt((x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2)
```


In this function, `A` and `B` are the two points, and `x`, `y`, and `z` are the coordinates of the points. The distance between the points is calculated by taking the square root of the sum of the squares of the differences between the `x`, `y`, and `z`.

It seemed to cut off the theory section:

 Here is an example of how you might finish the theory section for the `Distance3d` function:  

Theory

The distance between two points in 3d space is calculated using the Pythagorean theorem:

 Copy code

```
d = sqrt((x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2)
```

In this function, `A` and `B` are the two points, and `x`, `y`, and `z` are the coordinates of the points. The distance between the points is calculated by taking the square root of the sum of the squares of the differences between the `x`, `y`, and `z` coordinates of the points.

This formula is implemented in the `Distance3d` function by subtracting the `x`, `y`, and `z` coordinates of point `A` from the corresponding coordinates of point `B`, and then taking the square of each of these differences. These squares are then added together and the square root of the sum is taken to obtain the distance between the two points.

This method of calculating the distance between two points in 3d space is known as the Euclidean distance, and it is a common method for finding the distance between two points in any number of dimensions.

Note that like in previous examples, the 5.830 figure is incorrect. This was fixed, and some polish was added manually to create the file: Distance3d_AI.md