Exercise 5
Classification and Regression

Distance measurements

Euclidean distance	$d(A, B) = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2}$
Manhattan distance	$d(A, B) = \sum_{i=1}^{n} \lvert A_i - B_i \rvert$
Hamming distance	$d(A, B) = \#\{ i : A_i \neq B_i \}$

k-NN Algorithm:

```
Inputs: D: Dataset; k: number of neighbors to consider; Inst: the instance to classify;
Outputs: y: the class of the instance to classify
Var
    dist: array of [1..N] of pair (instance, distance); //with N the size of D = |D|
    knn: array of [1..k] of instances;
Begin
    For each instance X of D do
        dist[X] ←Calculate the distance between X and Inst;
    Done;
    dist ←Sort dist in ascending order of distances;
    knn ←the first k instances of dist;
    y ←The dominant class in knn;
return y;
End.
```

Questions :

Let's consider "*Exercise*" as the class (label) to predict.

1- Write a function to calculate the distance between two instances of the dataset (combine the Manhattan and the Hamming distances)

2- Write a function to sort the instances of the dataset according to the value of a calculated distance "d".

3- Write a function to return the dominant class among a set of K classes.

4- Implement the Knn algorithm.

5- Deduce the class of the instance

X = <2024-11-20 18:09:51.000, -0.137, 1.066, 0.8215, -6.597, 0.808, 1.985, B, medium, 30> with K = 3 then with K = 10.

Have fun !