Lab 6 - Modular Programming

1. Create a module that defines the following operations on arrays of integers: minimum, maximum, average, median. ★
2. Create a module that implements sets and the operations on sets (union, intersection, difference). ★★
3. Create a module that implements the K nearest neighbors algorithm. Given:
   - S: a set of n 2d points identified by their x and y coordinates and a letter l (label).
   - p: another 2d point that has no label
   - k: a positive integer

   Find the closest k points from S to p and return the label most commonly found among them.

   Example:

   - p = (x=1, y=1)
   - $k = 3$
   - S = {(x=0, y=0, l='r'), (x=0, y=1, l='r'), (x=-1.5, y=1.5, l='r'), (x=1.2, y=1.2, l='b'), (x=2, y=1, l='b')}

   The closest $k = 3$ points from S to p are {(x=0, y=1, l='r'), (x=1.2, y=1.2, l='b'), (x=2, y=1, l='b')}. In these 3 points the label 'b' appears 2 times and the label 'r' appears 1 time, so the result is 'b' (the label with the most apparitions).

   Hint: In order to compute the distance between 2 points use the Euclidean distance:

   $$d(a, b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

   Hint: In order to store the points you can use either an array of structs or a matrix of $n \times 3$ floats, where `m[i][0]` is the x coordinate of the i'th point, `m[i][1]` is the y coordinate of the i'th point, etc. ★★★

References

- Pb. 2 [1]

[1] Iosif Ignat & Marius Joldoș. *CP Laboratory Guide 6: Modular programming.*