151228619/151248619: OBJECT ORIENTED PROGRAMMING I
2022-2023 FALL SEMESTERS
LAB FINAL
Due: January 5, 2023, Thursday, 17:00

Remarks:
- Source code should be cleverly commented.
- You should write a report explaining design and implementation issues as well as anything a person may need to know while testing your program.
- You are expected to make a good design and exploit object oriented programming principles (encapsulation, composition, inheritance, STL, exception handling, etc.) as often as possible.
- For lab final, you need to send the project workspace including your .cpp and .h files. You must organize all this material in a single .zip or .rar archive, name the file as "yourID_Name.zip/rar", and load ESOGU UZEM.
- Grades will be given according to
    - 50% Class design and availability of object-oriented programming principles,
    - 15% code quality (commented and well formatted),
    - 15% documentation,
    - 20% functionality (implemented and correctly running functionalities).

Assume that we have **2 features** and **3 categories**. The features could be between **-5 and 5**. Also, we have **30 training samples** and each sample has an index, a category id, and 2 features. Index attribute of a sample is an identity number between **0 and 29**. Category attribute of a sample is a number between **0 and 2**. The first 10 samples must belong to Category 0, then, the second 10 samples must belong to Category 1, and the last 10 samples must belong to Category 2. The features of a sample must be between **-5 and 5**. Assign features randomly corresponding to that interval.
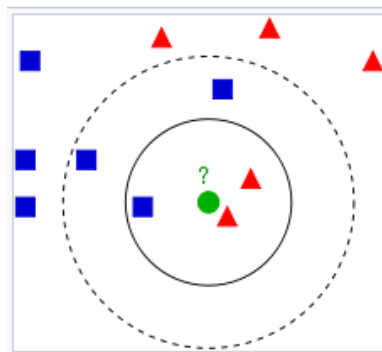

Fig 1. An example classification for K-NN algorithm. (Source: Wikipedia) The test sample (green dot) should be classified either to blue squares or to red triangles. If k = 3 (solid line circle) it is assigned to the red triangles because there are 2 triangles and only 1 square inside the inner circle. If k = 5 (dashed line circle) it is assigned to the blue squares (3 squares vs. 2 triangles inside the outer circle) [1].

We have also **10 test samples** with randomly assigned features between **-5 and 5.** Index attribute of a test sample is an identity number between **30 and 39.** In this lab final, you will design a class hierarchy to classify test samples with K-Nearest Neighbors classification algorithm (k-NN). **Use two separate containers for training and test samples.** For each test sample, employ **Euclidean distance** and **a plurality vote of its neighbor's criterion** to determine category of a test sample.

Your design must meet the following requirements:

 i) Data members of all classes must be private.
ii) Each class must contain constructor and destructor. Also, use member initializer and base class member initializer in appropriate situations.
iii) Validate data members if necessary. Use exception handling statements.
iv) Implement and use get and set functions for data members.
v) Implement appropriate access (read file and print functions) and utility functions.
vi) Each class must contain appropriate print functions or overloaded stream insertion operator (<<). Also, in appropriate situation you can implement operator overloading. This will get you extra credits.
vii) Use const qualifier, static storage class, and reference parameters in appropriate situations.
viii) Use dynamic memory allocation or use at least one STL container and one STL algorithm.
 ix) Use composition and/or inheritance in appropriate situations.
 x) The lab final is a console-based work but if you can provide graphical results such as Figure 1, you will get extra credits (20 pts).

[1] https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm