

# Assignment 01

## Part A:

Assume that you are given a text file named “short\_data.txt”. Each line except the first one of the data file contains information corresponding to a sample. The first value is a number that represents the value of an attribute X of a sample and the **second value is the classification (here the classification values are ‘YES’ and ‘NO’)**, It can be anything, which represents if the sample belongs to a class or not.

1. Now write a function named **readAllData** that takes as parameter the name of the file. The function then reads the data from the file and returns a **list of tuples** where each tuple corresponds to the information of a Sample.
2. Write a function named **computeAverageForClasses** to take the data read in function **readAllData** as argument and compute the average value of X for each class. The function then returns a dictionary where the keys are name of the class and the values are average of X for that class.
3. We assume that a sample should be a member of class A, if the value of X for that sample is closer to the average value of X for class A than any other classes. Now using this rule write a function named **misclassified** that returns the samples in this data is misclassified.
4. Using these functions print the average values of X for each class and the number of misclassified samples in the data. Finally, write the misclassified samples in a file named **“Misclassified.txt”**.

Task list:

1. Read the task description carefully.
2. Check the short\_data.txt files
3. Open functions.py
4. complete those functions:
  - a. **readAllData**: read all data from file and returns a list of tuples where each tuple corresponds to the information of a Sample.  
**for short\_data.txt ,readAllData will return**

```
[ ('15.79311545', 'YES'),  
  ('10.95644178', 'YES'),  
  ('27.77413952', 'NO'),  
  ('18.41013616', 'NO'),  
  ('18.74485271', 'NO'),  
  ('22.44869209', 'NO'),  
  ('18.09242441', 'NO'),  
  ('15.24850737', 'NO'),  
  ('11.21480959', 'YES'),  
  ('13.195037', 'NO')  
]
```

- b. **computeAverageForClasses**:  
**for short\_data.txt ,computeAverageForClasses will return**  
{'YES': 12.654788940000001, 'NO': 19.13054132285714}
  - c. **misclassified**  
**for short\_data.txt ,misclassified will return**  
[( '15.24850737', 'NO'), ( '13.195037', 'NO')]

Test:

1. Run test\_assignment.py file to see test results
2. **Don't modify test\_assignment.py and other file**
3. **You can only modify functions.py**
4. Run solutions.py you will find a file named "Misclassified.txt". It will be same as "Misclassified\_demo.txt" for short\_data.txt
5. Ignore **complete\_data.txt** and **complete\_data\_final.txt**. They are for testing purpose.  
**Don't remove them.**