**PART A (Binary)**

Phasor Measurement Unit (PMU) / Synchrophasor is a device which measures the electrical waves on an electricity grid, using a common time source for synchronization. From each PMU, there are 29 types of measurements. We have 4 PMUs in the system which measure 29 features totaling to 116 PMU measurement columns.

Training Data (TrainingDataBinary.csv)

* 6000 rows × 129 columns
* 6000 system traces, 50% normal events (0), 50% abnormal data injection attack events (1)
* In one trace, 128 features. Last column is event label (0/1).
  + 116 PMU measurements: indexed R#-Signal Reference (# is PMU number)
  + 12 control panel logs, Snort alerts and relay logs of the 4 PMUs

Testing Data (TestingDataBinary.csv)

* 100 rows × 128 columns
* No event labels

Work

* Design and implement ML algorithm to model the training data and compute labels (0, 1) for testing data
* Store labelled testing data (same columns as training data) in a file named TestingResultsBinary.csv and submit it with source code
* Report
  + Clear Problem Description
  + ML Technique used to compute labels
  + Clearly indicate the computed label for each trace in the testing data. You need to show the computed labels for all testing data – in the same order as was given to you
  + Analyse and discuss your results in terms of training error and training accuracy
    - Accuracy on testing data
    - Accuracy on training data

**PART B (Multi)**

Three types of events are evenly distributed: normal events (0), abnormal data injection attack events (1), and abnormal command injection attack events (2).

Training Data (TrainingDataMulti.csv)

* 6000 rows × 129 columns
* 6000 system traces, 128 features, 1 event label (0, 1, 2)

Testing Data (TestingDataMulti.csv)

* 100 rows × 128 columns
* No event labels

Work

* Design and implement ML algorithm to model the training data and compute labels for testing data (0,1,2)
* Store labelled testing data (same columns as training data) in a file named TestingResultsMulti.csv and submit it with source code
* Report
  + Clear Problem Description
  + ML Technique used to compute labels
  + Clearly indicate the computed label for each trace in the testing data. You need to show the computed labels for all testing data – in the same order as was given to you
  + Analyse and discuss your results in terms of training error and training accuracy
    - Accuracy on testing data
    - Accuracy on training data