Exercise 1 (50%)

a) Experimental setup can be the execution time of the system's sub-modules. So it is specified that execution time should be maximum 20ms and through the experimental setup we would like to see if this can be valid.

b) Variable that is measured is execution time.

c) Confounding variables are video frames and coordinates of the detected face.

d) I would redesign my algorithm in such way that execution time of system's sub-modules is maximum 20ms, for example I would select camera to capture images at less than 25 frames per second, for example at 23-24 images per second. Thus the reduction would be as less as possible so that there is no big deflection from the specifications. The same would be done if there was effect from writing data (coordinates of the detected face), i.e. I would reduce the frequency of writing coordinates per second, so that specifications are met.

I) The sample mean of the sample test [$t_{1,}$.....t_{36}] is $\acute{x}=¿$20.86=21 which is something greater than 20ms

II) The standard deviation of the sample test [$t_{1,}$.....t_{36}] is σ=1.74=2.

III) Null hypothesis: Execution time of the system's sub-modules is maximum 20ms

IV) Null hypothesis that was stated above is not retained due to the fact that sample mean of the sample test is 20.86>20ms.Thus an alternative hypothesis can be that execution time of the system's sub-modules is maximum 21ms.

V) To test the null hypothesis using the test statistic at a 99% confidence level means that we must test the hypothesis using significance level a=0.01.In that case using one-tailed, in order null hypothesis to be rejected there must be z< 2.326 or z< -2.326.So we take that z must be lower than 2.326.Below this value are critical values for which our hypothesis is rejected. So if z doesn't belong in the critical values' interval then our null hypothesis retains. Computing z we have,

z= $\frac{\acute{\chi}-\mu}{\frac{\sigma}{\sqrt{n}}}$, where $\acute{x}$=21, μ=20, σ=2, n=36

Making the calculations we find that z=3.Therefore z>2.326, therefore our null hypothesis retains.

VI) My task here was to design and develop the face detection sub-module. I was given the following specifications: 1) The selected camera captures images at 25 frames per second, 2) The video acquisition and pre-processing module has a maximum execution time of 10ms, 3)

The face tracking and the decision making and alerting subsystems together have a maximum execution time of 10ms, 4) My subsystem (algorithm) should read data (video frames) and writes (the coordinates of the detected face) to volatile memory.

So according to these given instructions I made experimental setup of my subsystem's algorithm to see if it's executed in the instructed time (20ms). So I took the following set of data (times in milliseconds): [$t_{1,}$ … t_{12}¿ = [20.8, 21.0, 18.0, 19.0, 18.3, 19, 24.0, 22.3, 21.5, 22.8, 21.6, 20.5], [$t_{13,}$ …t_{24}] = [22.0, 22.0, 19.5, 19.0, 24.0, 18.0, 20.4, 20.8, 25, 18.9, 20.5, 21.3], [$t_{25,}$ …t_{26}] = [19.1, 22.3, 21.4, 21.5, 22.8, 21.6, 20.5, 19.0, 18.9, 22.0, 21.3,20.4].

From these data after calculations we see that the mean execution time that our system performs is 21ms.So it is greater than 20ms but very close. After testing the ordered specification (maximum execution time in 20ms) being confident at 99% sure that our system can be executed in the specified time, I see that my implementation of the face recognition sub-module guarantees the real time performance of the whole system at 99%.So there is some possibility just 1% that time can exceed the specified execution time.

Exercise II (50%)

In order to find out which algorithm got better performance it is necessary to use the following equations.

<u>For the first algorithm:</u>

	Actual Class	Predicted Class
	Negative	Positive
Negative	2725	35
Positive	95	145

<u>For the second algorithm:</u>

	Actual Class	Predicted Class
	Negative	Positive
Negative	2710	50
Positive	90	150

For the calculation of Accuracy is used the equation below:

$$\mathrm{A}(M) = \frac{TN + TP}{TN + FP + FN + TP}$$

Algorithm I: Accuracy = (2725 + 145) / 2725 + 35 + 95 + 145 = 0,92

Algorithm II: Accuracy = (2710 + 150) / 2710 + 50 + 90 + 150 = 0,95

For the calculation of Recall is used the equation below:

$$\text{Recall} = \frac{TP}{TP + FN}$$

Algorithm I: Recall = 145 / (145 + 90) = 0,60

Algorithm II: Recall = 150 / (150 + 90) = 0,63

For the calculation of Precision is used the equation below:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Algorithm I: Precision = 145 / (145 + 35) = 0,81

Algorithm II: Precision = 150 / (150 + 50) = 0,75

For the calculation of F-mesure is used the equation below:

$$\text{F} = \frac{2}{\left(\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}\right)}$$

Algorithm I: F-mesure = 2 * (0,81 * 0,60) / (0,81 + 0,60) = 0,69

Algorithm II: F-mesure =2 * (0,75 * 0,63) / (0,75 + 0,63) = 0,68