**Documentation of Monitored Assignment 2**

Description of the algorithm:

The algorithm consists of 8 classes («Main», «Pixel», «PaintPane», «Controller», «Patterns», «PaintException», «FFBP», «FFBPError» ).

The main idea of the programm is to make the stage with a grid and fill the grid with the initial (white) color squares (pixels). Clicking on one of the squares on the stage this particular pixel changes its color (it becomes black). If we draw the letter the programm should show us in the bar chart which letter our symbol looks like(we programmed the letter from „A“ to „H“). Cliking on the button with the letter we receive the picture of the ideal kind of such letter.

In the class «Main»(extended from JavaFX class called «Application») we set all the parameters of the stage (space, paddings and so on). The class «PaintPane» draws the pixels and clears the field when we click the right button of the mouse (class «Pixel» and «PaintExceptions» are included here).

The main class which executes the «research» of the drawn letter is a class «Controller». This class accepts the object which was drawn (data type «PaintPane»), corrects it according to the class «FFBP» and creates FFBP-object (using class «Patterns») in accordance with the task (setting layout, using method «randomize», «setEta» etc.). Then this method assigns a controller (we assign the controller in order to reach BarCharts updates and apply them then we draw pixels), creates buttons, events for them and the bar chart.

When we click on the button «New Net», the new net (with the method «getNewNet») is created and the method «updateBarChartSeries» is applied accepting series of the chart.

After clicking «Learn 500 Cycles» we launch method «learn» with number of cycles (in our case 500) and update the bar chart with the method «updateBarChartSeries» with series. This method checks if Noise is active (method «getMatrixWithNoise» returns us the matrix after processing one in the method «applyNoise») and applies matrix with noise or without noise. Two methods of the class «FFBP» are applied for ours net:

-«activateInputAndFeedForward» (we apply the returned vector of ours martx after usind the method «getVector»)

-«applyDesiredOutputAndPropagateBack».

When we click on the button with the name of the letter without noise, we receive the letter which is drawn according to the matrix from the method «Patterns». With noise there is a letter with an 10% deviation from the ideal letter. The method «paintByMatrix» is launched with the result of the method «getMatrixWithNoise», accroding to the result of this method it takes the pattern and draws us the letter.

|  |
| --- |
| Group list: |
| Anton Gorshkov |
| Rodion Danilenko |
| Henri Cela |