Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 2а

з дисципліни «Методи наукових досліджень»

на тему «ДОСЛІДЖЕННЯ НЕЙРОННИХ МЕРЕЖ. МОДЕЛЬ PERCEPTRON»

Виконав:

студент ІІ курсу ФІОТ

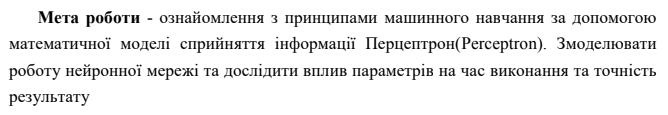
групи ІВ-93

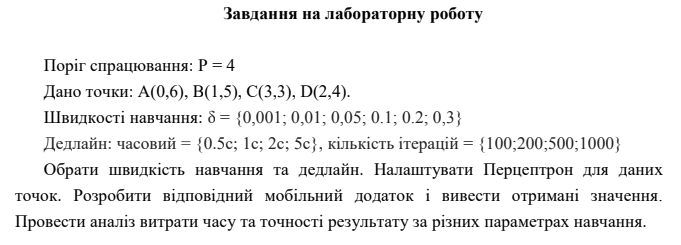
Ільяш В. В.

ПЕРЕВІРИВ:

ас. Регіда П. Г.

Київ - 2021





**Код програми:**

package com.example.mndlab2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.TextView;

import com.google.android.material.chip.Chip;

import com.google.android.material.chip.ChipGroup;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

public void click(View view) {

TextView tw=findViewById(R.id.textView4);

ChipGroup cg1 = findViewById(R.id.cg1);

Chip c1 = findViewById(cg1.getCheckedChipId());

double Speed\_Of\_Training = Double.parseDouble(c1.getText().toString());

ChipGroup cg2 = findViewById(R.id.cg2);

Chip c2 = findViewById(cg2.getCheckedChipId());

double Time\_Limit = Double.parseDouble(c2.getText().toString());

ChipGroup cg3 = findViewById(R.id.cg3);

Chip c3 = findViewById(cg3.getCheckedChipId());

double Number\_Of\_Iterations = Double.parseDouble(c3.getText().toString());

int Current\_Iteration=0;

int P = 4;

int Succesful=0;

double W1 = 0;

double W2 = 0;

int[][] points = {{0, 6}, {1, 5}, {3, 3}, {2, 4}};

long start = System.nanoTime();

while((((System.nanoTime()-start)/Math.pow(10,9))<Time\_Limit)&&

(Current\_Iteration<Number\_Of\_Iterations)&&(Succesful<4)){

int Point\_Index=Current\_Iteration%4;

double y=W1\*points[Point\_Index][0]+W2\*points[Point\_Index][1];

if (((y>P)&&(Point\_Index<2))||((y<P)&&(Point\_Index>1))){

Succesful++;

Current\_Iteration++;

continue;

}

double Delta=P-y;

W1=W1+Delta\*points[Point\_Index][0]\*Speed\_Of\_Training;

W2=W2+Delta\*points[Point\_Index][1]\*Speed\_Of\_Training;

Current\_Iteration++;

Succesful=0;

}

tw.setText("W1="+W1+"\nW2="+W2+"\nЧас виконання="

+(System.nanoTime()-start)+" нс");

}

}

**Результати роботи програми:**

