

Міністерство освіти і науки України  
Львівський національний університет імені Івана Франка

**Звіт**  
про виконання лабораторної роботи  
**«Побудова функцій приналежності нечіткої множини на основі експертної інформації»**

Виконав:  
студент групи ФЕІ – 54  
Васюк Я.В.  
Перевірів:  
Сінькевич О.О.

Львів 2021

## Лабораторна робота №1

### Побудова функцій приналежності нечіткої множини на основі експертної інформації

**Мета роботи:** ознайомитися з прямим методом побудови функцій приналежності нечіткої множини, який ґрунтується на статистичній обробці думок групи експертів.

#### Завдання:

1. На основі опитування 5 або більше експертів задати нечітку множину «чоловік середнього зросту». Для цього кожен експерт заповнює опитувальник, в якому вказує свою думку про наявність у елементів  $x_i$ ,  $i = [1, n]$  властивостей нечіткої множини  $A_j$ ,  $j = [1, m]$ . Опитувальник має вигляд таблиці (у випадку нечіткої множини «чоловік середнього зросту»):

	$x_1$ (150 см)	$x_2$ (155 см)	...	$x_n$ (200 см)
$b_{i^1, j}$				
$b_{i^2, j}$				
...				
$b_{i^k, j}$				

У таблиці використані наступні позначення:  $k$  – кількість експертів;  $b_{i^k, j}$  – думка  $k$ -го експерта про наявність у елемента  $x_i$  властивостей нечіткої множини

$A_j$ . У випадку бінарних оцінок експертів, тобто  $b_{i^k, j} \in \{0, 1\}$ , одиниця (нуль) вказує на наявність (відсутність) у елемента  $x_i$  властивостей нечіткої множини  $A_j$ . За результатами опитування експертів, ступінь приналежності елемента  $x_i$  нечіткій множині  $A_j$  розраховують наступним чином:

$$A_j(x_i) = \frac{1}{k} \sum_{k=1}^k b_{i^k, j}$$

У випадку субнормальної нечіткої множини її нормалізують за допомогою виразу:

$$A_j(x_i) = \frac{A_j(x)}{\sup A_j(x)}$$

2. Створити програму, яка розраховує функцію приналежності на основі статистичної обробки думок групи експертів, буде її графік і апроксимує однією з  $\Pi$ -го експерта про наявність у елемента подібних функцій.

3. Визначити носій нечіткої множини, її ядро та границі.

### Виконання:

Було розроблено програму в Windows Forms, яка складається із двох форм (Form1, Form4).

150 см	0	180 см	1
155 см	0	185 см	1
160 см	0	190 см	1
165 см	0	195 см	1
170 см	q	200 см	1
175 см	1		

Оцініть, чи належить кожен елемент до множини "Високий чоловік" (0 - ні, 1 - так)

Прийнят      Оновит      Далі

Рис.1. Form 1

Form1 представляє собою форму для опитування експертів, кожен експерт визначає чи належить елемент до множини та натискає «прийняти». За допомогою кнопки «оновити», відповідно можна оновити результати опитування експерта. Кнопка «далі» слугує для переходу в Form4.

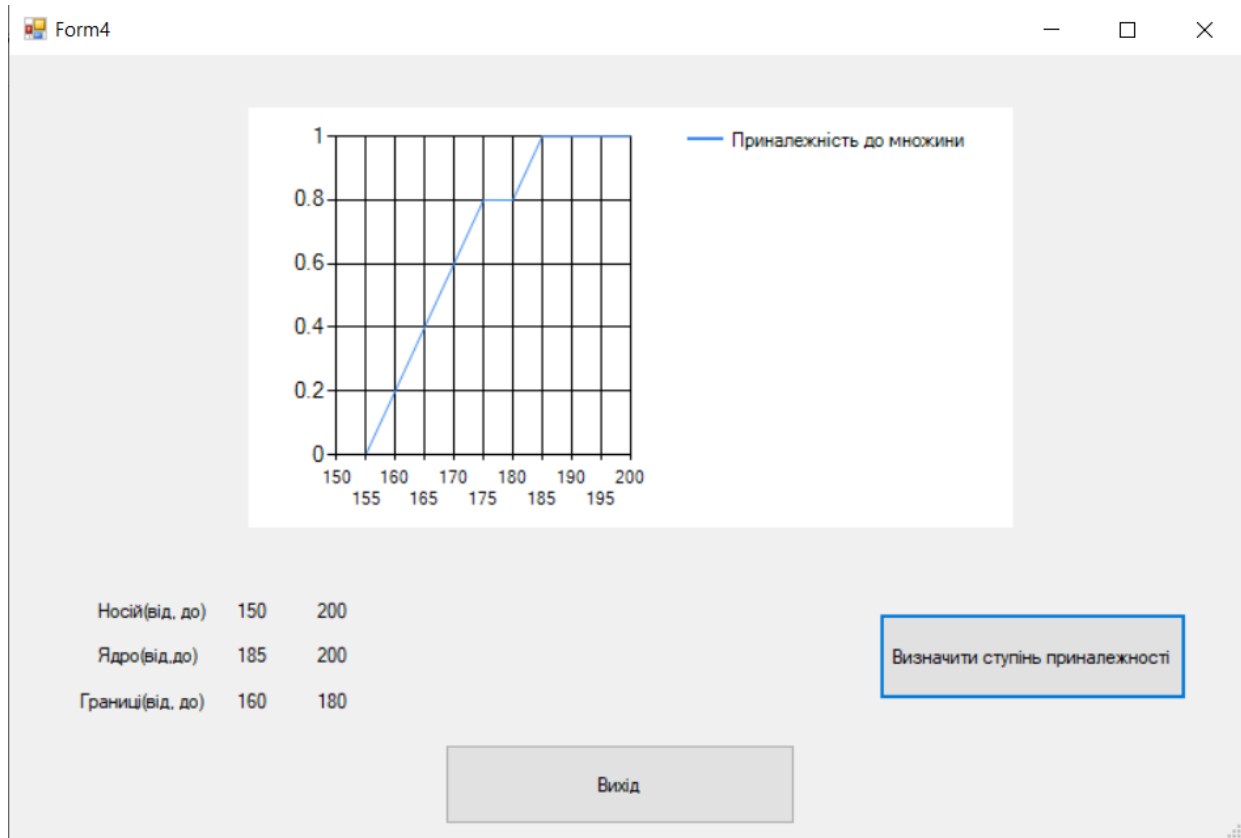


Рис.2. Form4

Form4 являє собою форму для візуалізації результатів виконання завдань лабораторної роботи. Кнопка «Визначити стіпунь приналежності» будує графік функції приналежності, а також визначає носій, ядро та гранці нечіткої множини на основі даних із Form1

**Висновок:** Я ознайомився з прямим методом побудови функцій приналежності нечіткої множини, який ґрунтується на статистичній обробці думок групи експертів.

**Код:**

### Form1:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace lab1 olenych
{
    public partial class Form1 : Form
    {
        double[,] arr = new double[10, 11];
        int count = 0;
        public Form1()
        {
```

```

        InitializeComponent();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        Form4 fr4 = new Form4(arr);
        fr4.ShowDialog();
        this.Close();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        arr[count,0] = Convert.ToDouble(textBox2.Text);
        arr[count,1] = Convert.ToDouble(textBox3.Text);
        arr[count,2] = Convert.ToDouble(textBox4.Text);
        arr[count,3] = Convert.ToDouble(textBox5.Text);
        arr[count,4] = Convert.ToDouble(textBox6.Text);
        arr[count,5] = Convert.ToDouble(textBox7.Text);
        arr[count,6] = Convert.ToDouble(textBox1.Text);
        arr[count,7] = Convert.ToDouble(textBox8.Text);
        arr[count,8] = Convert.ToDouble(textBox9.Text);
        arr[count,9] = Convert.ToDouble(textBox10.Text);
        arr[count,10] = Convert.ToDouble(textBox11.Text);
        count++;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        textBox1.Text = null;
        textBox2.Text = null;
        textBox3.Text = null;
        textBox4.Text = null;
        textBox5.Text = null;
        textBox6.Text = null;
        textBox7.Text = null;
        textBox8.Text = null;
        textBox9.Text = null;
        textBox10.Text = null;
        textBox11.Text = null;
    }
}
}

```

## Form4:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace lab1.olenych
{
    public partial class Form4 : Form
    {
        double [] arr4 = new double [11];
        double[,] arr1 = new double[10, 11];
        public Form4(double [,] arr)
        {

```

```

        InitializeComponent();
        arr1 = arr;
    }

    private void button1_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    public void button2_Click(object sender, EventArgs e)
    {
        for (int i = 0; i != 11; i++)
        {
            arr4[i] = (arr1[0, i] + arr1[1, i] + arr1[2, i] + arr1[3, i] + arr1[4, i])
/ 5;
        }
        int[] arr5 = new int[] { 150, 155, 160, 165, 170, 175, 180, 185, 190, 195, 200
    };

        int y = 0;
        for (int i = 10; i >= 0; i--)
        {
            if (arr4[i] == 0)
            {
                label11.Text = Convert.ToString(arr5[i]);
            }
        }
        label4.Text = Convert.ToString(arr5[10]);
        for (int i = 0; i != 11; i++)
        {
            if (arr4[i] == 1)
            {
                label12.Text = Convert.ToString(arr5[i]);
                break;
            }
        }
        label5.Text = Convert.ToString(arr5[10]);
        for (int i = 10; i >= 0; i--)
        {
            if (arr4[i] != 0 && arr4[i] != 1)
            {
                label6.Text = Convert.ToString(arr5[i]);
                break;
            }
        }
        for (int i = 0; i != 11; i++)
        {
            if (arr4[i] != 0 && arr4[i] != 1)
            {
                label3.Text = Convert.ToString(arr5[i]);
                break;
            }
        }
        chart1.ChartAreas[0].AxisX.Minimum = 150;
        chart1.ChartAreas[0].AxisX.Maximum = 200;
        chart1.ChartAreas[0].AxisY.Minimum = 0;
        chart1.ChartAreas[0].AxisY.Maximum = 1;
        for (int i = 0; i != 11; i++)
        {
            chart1.Series["Приналежність до множини"].Points.AddXY(arr5[i], arr4[i]);
        }
    }
}

```

} }