

Національний технічний університет України  
«Київський політехнічний інститут»  
Факультет інформатики і обчислювальної техніки  
Кафедра обчислювальної техніки

Лабораторна робота №1  
З алгоритмів та методів обчислень  
Варіант 25

*Виконав:*  
Студент групи ІО-32  
Попенко Руслан  
*Перевірив:*  
Порєв В. М.

Київ - 2015 р.

### 1. Тема завдання:

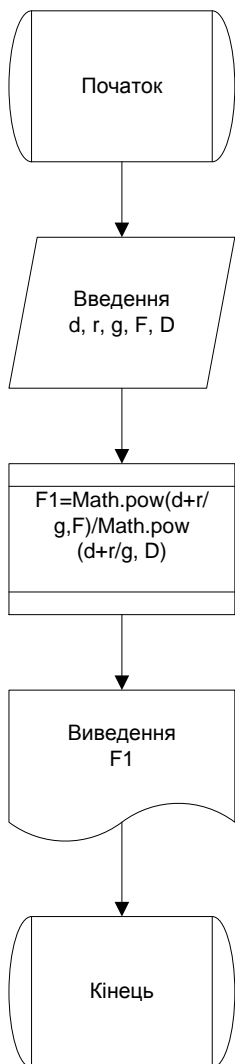
Відповідно до варіанту завдання розробити блок-схеми обчислення виразів для лінійного алгоритму, алгоритму, що розгалужується та циклічного алгоритму. У відповідності до блок-схеми створити програму обчислення виразу на алгоритмічній мові Pascal.

### 2. Завдання:

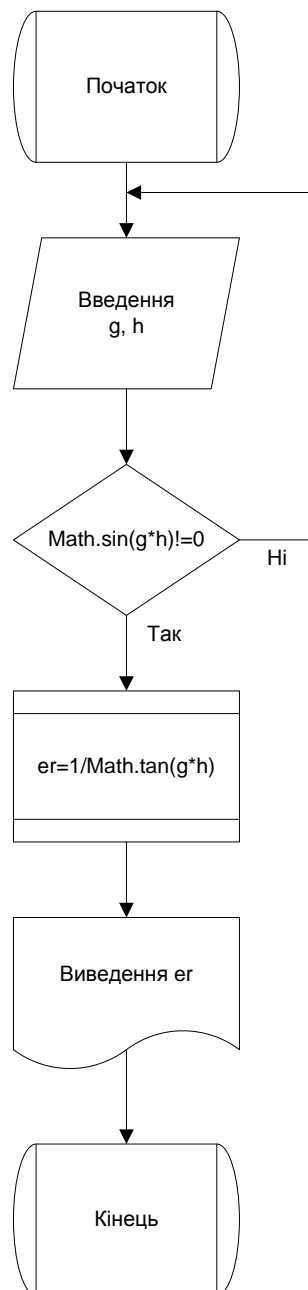
Лінійний	Що розгалужується	Циклічний
$F1 = (d+r/g)^F / (d+r/g)^D$	Обчислити значення $er = \text{ctg}(gh)$	Обчислити $J = \frac{1}{n_1} + \frac{1}{n_2} + \dots + \frac{1}{n_m}$ $m=25$

### 3. Блок-схеми алгоритмів:

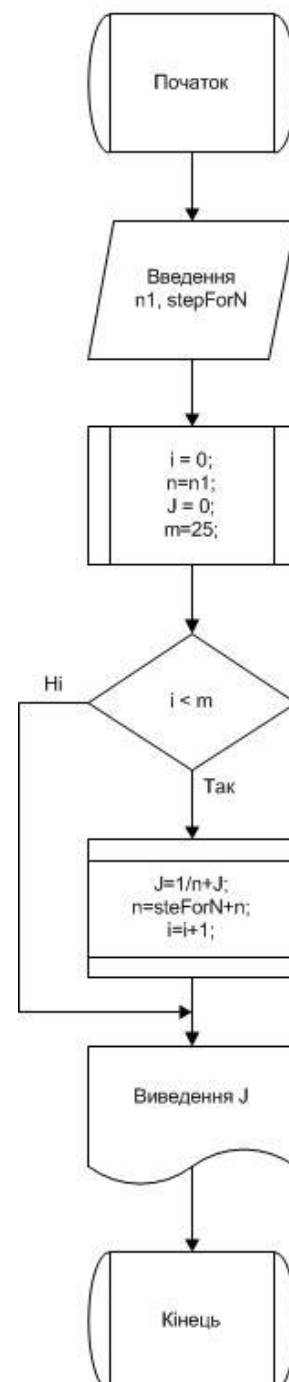
#### Лінійний



#### Що розгалужується



#### Циклічний



#### 4. Лістинг програми:

##### Клас головного вікна

```
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.JTextArea;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JTextField;
import javax.swing.JLabel;
import java.awt.Font;
import javax.swing.JMenu;
import javax.swing.JRadioButtonMenuItem;
import java.awt.CardLayout;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.io.PrintWriter;

public class lab1 {

    private JFrame frame;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    lab1 window = new lab1();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    public static void write(String fileName, String text) {
        //Определяем файл
        File file = new File(fileName);

        try {
            //проверяем, что если файл не существует то создаем его
            if(!file.exists()){
                file.createNewFile();
            }

            //PrintWriter обеспечит возможности записи в файл
            PrintWriter out = new PrintWriter(file.getAbsolutePath());

            try {
                //Записываем текст у файл
                out.print(text);
            } finally {
                //После чего мы должны закрыть файл
                //Иначе файл не запишется
                out.close();
            }
        } catch (IOException e) {
            throw new RuntimeException(e);
        }
    }

    public static String read(String fileName) throws FileNotFoundException {
        //Этот спец. объект для построения строки
        StringBuilder sb = new StringBuilder();

        exists(fileName);

        try {
            File file =new File(fileName);
            //Объект для чтения файла в буфер
            BufferedReader in = new BufferedReader(new FileReader( file.getAbsolutePath()));
            try {
                //В цикле построчно считываем файл
                String s;
```

```

        while ((s = in.readLine()) != null) {
            sb.append(s);
            sb.append("\n");
        }
    } finally {
        //Также не забываем закрыть файл
        in.close();
    }
} catch(IOException e) {
    throw new RuntimeException(e);
}

//Возвращаем полученный текст с файла
return sb.toString();
}

private static void exists(String fileName) throws FileNotFoundException {
    File file = new File(fileName);
    if (!file.exists()){
        throw new FileNotFoundException(file.getName());
    }
}

/**
 * Create the application.
 */
public lab1() {
    initialize();
}

/**
 * Initialize the contents of the frame.
 */
private void initialize() {
    frame = new JFrame();
    frame.setTitle("\u0418\u0430\u0431\u0435\u0440\u0430\u0442\u0435\u0440\u043d\u0430 \u21161
\u041f\u0435\u043f\u0435\u043d\u0430\u0430\u0430 \u0420\u0443\u0441\u043b\u0430\u0430\u0430.
\u0412\u0430\u0440\u0440\u0430\u0430\u0430\u0430\u0430 25");
    frame.setBounds(100, 100, 632, 372);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.getContentPane().setLayout(null);

    JButton button = new JButton("\u0418\u0430\u0430\u0430\u0430\u0430\u0430\u0430 ");
    button.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {

            liner LineAlg=new liner();
            LineAlg.setVisible(true);

        }
    });
    button.setFont(new Font("Times New Roman", Font.BOLD, 25));
    button.setBounds(35, 74, 239, 66);
    frame.getContentPane().add(button);

    JButton button_1 = new JButton("\u0420\u0430\u0430\u0430\u0430\u0430\u0430\u0430");
    button_1.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {

            Cyclic cycl=new Cyclic();
            cycl.setVisible(true);

        }
    });
    button_1.setFont(new Font("Times New Roman", Font.BOLD, 25));
    button_1.setBounds(321, 74, 239, 66);
    frame.getContentPane().add(button_1);

    JButton button_2 = new JButton("\u0420\u0435
\u0440\u0435\u0430\u0430\u0430\u0430\u0430\u0430\u0430\u0430\u0430\u0430\u0430\u0430");
    button_2.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            branchingOut brancOut=new branchingOut();
            brancOut.setVisible(true);

        }
    });
    button_2.setFont(new Font("Times New Roman", Font.BOLD, 25));
    button_2.setBounds(155, 176, 287, 88);
    frame.getContentPane().add(button_2);

    JLabel label = new JLabel("");
    label.setBounds(134, 11, 0, 0);

```

## Клас вікна лінійного алгоритма

```
DtextField = new JTextField();
```

```

DtextField.setBounds(106, 75, 86, 20);
contentPane.add(DtextField);
DtextField.setColumns(10);

JLabel lblR = new JLabel("r =");
lblR.setFont(new Font("Times New Roman", Font.BOLD, 18));
lblR.setBounds(202, 68, 22, 22);
contentPane.add(lblR);

RtextField_1 = new JTextField();
RtextField_1.setColumns(10);
RtextField_1.setBounds(236, 75, 86, 20);
contentPane.add(RtextField_1);

JLabel lblG = new JLabel("g =");
lblG.setFont(new Font("Times New Roman", Font.BOLD, 18));
lblG.setBounds(332, 68, 24, 22);
contentPane.add(lblG);

GtextField_2 = new JTextField();
GtextField_2.setColumns(10);
GtextField_2.setBounds(374, 75, 86, 20);
contentPane.add(GtextField_2);

JLabel lblF = new JLabel("F =");
lblF.setFont(new Font("Times New Roman", Font.BOLD, 18));
lblF.setBounds(106, 133, 25, 22);
contentPane.add(lblF);

FFtextField = new JTextField();
FFtextField.setBounds(141, 136, 86, 20);
contentPane.add(FFtextField);
FFtextField.setColumns(10);

JLabel lblD_1 = new JLabel("D = ");
lblD_1.setFont(new Font("Times New Roman", Font.BOLD, 18));
lblD_1.setBounds(272, 133, 33, 22);
contentPane.add(lblD_1);

DDtextField_1 = new JTextField();
DDtextField_1.setColumns(10);
DDtextField_1.setBounds(307, 136, 86, 20);
contentPane.add(DDtextField_1);

JButton btnNewButton = new JButton("\u0414\u0435\u0440\u0430\u0445\u0443\u0432\u0430\u0442\u0438");
btnNewButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        double d,r,g,F,D, res;
        try {
            d=Double.parseDouble(DtextField.getText());
            r=Double.parseDouble(RtextField_1.getText());
            g=Double.parseDouble(GtextField_2.getText());
            F=Double.parseDouble(FFtextField.getText());
            D=Double.parseDouble(DDtextField_1.getText());

            res=Math.pow(d+r/g, F)/Math.pow(d+r/g, D);
            REtextField.setText(""+res);
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "Перевірте правильність вводу даних");
        }
    }
});
btnNewButton.setFont(new Font("Times New Roman", Font.BOLD, 18));
btnNewButton.setBounds(186, 167, 150, 31);
contentPane.add(btnNewButton);

JLabel lblF_1 = new JLabel("\u0424\u0435\u0437\u0443\u0438\u0442\u0440\u0430\u0442\u0430\u0442 F1 =");
lblF_1.setFont(new Font("Times New Roman", Font.BOLD, 20));
lblF_1.setBounds(61, 232, 131, 24);
contentPane.add(lblF_1);

REtextField = new JTextField();
REtextField.setBounds(202, 236, 272, 22);
contentPane.add(REtextField);
REtextField.setColumns(10);

JButton button = new JButton("\u0417\u0430\u0434\u0440\u0438\u0442\u0438");
button.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        dispose();
    }
});

```

```

});
button.setBounds(419, 269, 123, 34);
contentPane.add(button);

JButton button_1 = new JButton("\u0417\u0430\u043F\u0438\u0441\u0430\u0442\u0438 \u0432\u0444\u0430\u0439\u0438\u0432");
button_1.setFont(new Font("Times New Roman", Font.BOLD, 15));
button_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        try {
            String d,r,g,F,D;
            d=DtextField.getText();
            r=RtextField_1.getText();
            g=GtextField_2.getText();
            F=FFtextField.getText();
            D=DDtextField_1.getText();
            String buf=d+" "+r+" "+g+" "+F+" "+D;
            String file_name="D://obuchenie/2 курс 4 семестр/Алгоритми та методи
обчислень/Лабораторна робота №1/liner.txt";
            lab1.write(file_name, buf);

        } catch (Exception e2) {
            JOptionPane.showMessageDialog(null, "Помилка");
        }
    }
});
button_1.setBounds(15, 166, 161, 36);
contentPane.add(button_1);

JButton button_2 = new JButton("\u0417\u0447\u0438\u0442\u0442\u0430\u0442\u0438 \u0456\u0437\u0442\u0438 \u0456\u0437\u0444\u0443\u0439\u0438\u0432");
button_2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        String str="";
        try {
            str=lab1.read("D://obuchenie/2 курс 4 семестр/Алгоритми та методи
обчислень/Лабораторна робота №1/liner.txt");
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
        String strVector [];
        strVector=str.split(" ");
        DtextField.setText(strVector[0]);
        RtextField_1.setText(strVector[1]);
        GtextField_2.setText(strVector[2]);
        FFtextField.setText(strVector[3]);
        DDtextField_1.setText(strVector[4]);
    }
});
button_2.setFont(new Font("Times New Roman", Font.BOLD, 15));
button_2.setBounds(362, 167, 161, 36);
contentPane.add(button_2);
}
}

```

## Клас вікна алгоритма, що розгалужується

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextField;
import javax.swing.JButton;

import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.io.FileNotFoundException;

public class branchingOut extends JFrame {

    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;

```

```

private JTextField textField_2;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                branchingOut frame = new branchingOut();
                frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the frame.
 */
public branchingOut() {
    setTitle("\u0041\u003B\u0033\u0043\u0048\u0042\u0043C, \u0049\u003E\u00437\u0033\u003B\u0043\u0043\u0043\u0045\u0042\u0044C\u0041\u0044F");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 502, 291);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);
    contentPane.setLayout(null);

    JLabel label = new JLabel("\u0042\u0035\u0039 \u0030\u003B\u0033\u0043\u0044\u0038\u0042\u003C\u0032\u0038\u0040\u0045\u0043\u0042\u0043\u0045 \u0043\u0047\u0030\u00447\u0035\u003D\u0044F\u0043\u0043\u0043\u0043\u0046\u00457 er=ctg(gh)");
    label.setFont(new Font("Times New Roman", Font.BOLD, 14));
    label.setBounds(61, 22, 383, 17);
    contentPane.add(label);

    JLabel lblG = new JLabel("g =");
    lblG.setFont(new Font("Times New Roman", Font.BOLD, 18));
    lblG.setBounds(62, 57, 24, 22);
    contentPane.add(lblG);

    textField = new JTextField();
    textField.setBounds(96, 60, 86, 20);
    contentPane.add(textField);
    textField.setColumns(10);

    JLabel lblH = new JLabel("h =");
    lblH.setFont(new Font("Times New Roman", Font.BOLD, 18));
    lblH.setBounds(216, 57, 25, 22);
    contentPane.add(lblH);

    textField_1 = new JTextField();
    textField_1.setColumns(10);
    textField_1.setBounds(266, 60, 86, 20);
    contentPane.add(textField_1);

    JButton btnNewButton = new JButton("\u0041F\u0043E\u0044\u0030\u0045\u0043\u0032\u0030\u0042\u0038");
    btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            double g,h, res=0;
            try {
                g=Double.parseDouble(textField.getText());
                h=Double.parseDouble(textField_1.getText());

                if (Math.sin(g*h)!=0.0){
                    res=1.0/Math.tan(g*h);
                } else {
                    int res=1/0;
                }

                textField_2.setText(""+res);
            } catch (Exception e) {
                JOptionPane.showMessageDialog(null, "Перевірте правильність вводу даних, cth(x), x!=pi*n");
            }
        }
    });
    btnNewButton.setFont(new Font("Times New Roman", Font.BOLD, 22));
}

```



```

        btnNewButton.setBounds(239, 96, 175, 35);
        contentPane.add(btnNewButton);

        JLabel lblNewLabel = new JLabel("er = ");
        lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 20));
        lblNewLabel.setBounds(96, 154, 39, 24);
        contentPane.add(lblNewLabel);

        textField_2 = new JTextField();
        textField_2.setBounds(145, 159, 230, 19);
        contentPane.add(textField_2);
        textField_2.setColumns(10);

        JButton button = new JButton("\u0417\u0430\u0434\u0440\u0438\u0442\u0438\u0432\u0456\u0434\u0430\u0434\u0435");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                dispose();
            }
        });
        button.setBounds(331, 203, 134, 22);
        contentPane.add(button);

        JButton btnNewButton_1 = new JButton("\u0417\u0430\u0434\u0438\u0438\u0441\u0430\u0442\u0438 \u0432\u0444\u0430\u0434\u0438\u0432");
        btnNewButton_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    String g="", h="";
                    g=textField.getText();
                    h=textField_1.getText();
                    String buf=g+" "+h;
                    String file_name="D://obuchenie/2 курс 4 семестр/Алгоритми та методи обчислень/Лабораторна робота №1/branchingOut.txt";
                    lab1.write(file_name, buf);

                } catch (Exception e2) {
                    JOptionPane.showMessageDialog(null, "Помилка");
                }
            }
        });
        btnNewButton_1.setBounds(46, 86, 136, 24);
        contentPane.add(btnNewButton_1);

        JButton button_1 = new JButton("\u0417\u0447\u0438\u0442\u0442\u0438\u0442\u0438 \u0456\u0434\u0438\u0442\u0438");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String str="";
                try {
                    str=lab1.read("D://obuchenie/2 курс 4 семестр/Алгоритми та методи обчислень/Лабораторна робота №1/branchingOut.txt");
                } catch (FileNotFoundException e3) {
                    e3.printStackTrace();
                }
                String strVector [];
                strVector=str.split(" ");
                textField.setText(strVector[0]);
                textField_1.setText(strVector[1]);
            }
        });
        button_1.setBounds(46, 124, 136, 24);
        contentPane.add(button_1);
    }
}

```

## Клас вікна циклічного алгоритма

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextField;
import javax.swing.JButton;

```

```
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.io.FileNotFoundException;
```

```
public class Cyclic extends JFrame {
```

```
    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;
    private JTextField textField_2;
```

```
    /**
     * Launch the application.
     */
```

```
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Cyclic frame = new Cyclic();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
```

```
    /**
     * Create the frame.
     */
```

```
    public Cyclic() {
        setTitle("\u0026\u0038\u003A\u003B\u0056\u0047\u003D\u0038\u0039\u0030\u003B\u0043E\u0040\u0038\u0042\u003C"");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 539, 247);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JLabel label = new JLabel("\u0026\u0035\u0039 \u0030\u003B\u0043\u0040\u0045\u0042\u0043\u0044\u0038\u0042\u0043C"");
        label.setFont(new Font("Times New Roman", Font.BOLD, 14));
        label.setBounds(28, 11, 414, 17);
        contentPane.add(label);

        JLabel lblN = new JLabel("n1 = ");
        lblN.setFont(new Font("Times New Roman", Font.BOLD, 18));
        lblN.setBounds(38, 39, 39, 22);
        contentPane.add(lblN);

        textField = new JTextField();
        textField.setBounds(131, 39, 126, 22);
        contentPane.add(textField);
        textField.setColumns(10);

        JLabel lblN_1 = new JLabel("\u0041\u0040\u0043E\u003A \u0034\u003B\u0044F n");
        lblN_1.setFont(new Font("Times New Roman", Font.BOLD, 18));
        lblN_1.setBounds(32, 72, 92, 22);
        contentPane.add(lblN_1);

        textField_1 = new JTextField();
        textField_1.setColumns(10);
        textField_1.setBounds(131, 74, 126, 22);
        contentPane.add(textField_1);

        JLabel lblNewLabel = new JLabel("J =");
        lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 18));
        lblNewLabel.setBounds(160, 106, 24, 22);
        contentPane.add(lblNewLabel);

        textField_2 = new JTextField();
        textField_2.setColumns(10);
        textField_2.setBounds(194, 107, 126, 22);
        contentPane.add(textField_2);

        JButton btnNewButton = new JButton("\u001F\u0043E\u0040\u0030\u0045\u0043\u0032\u0030\u0042\u0038");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
```

```

        double n1, n, step, J=0;
        int m=25;
        try {
            n1=Double.parseDouble(textField.getText());
            step=Double.parseDouble(textField_1.getText());
            n=n1;
            for (int i = 0; i < m; i++) {
                J+=1/n;
                n+=step;
            }
            textField_2.setText(""+J);
        } catch (Exception e2) {
            JOptionPane.showMessageDialog(null, "Перевірте правильність вводу даних");
        }
    }

});
btnNewButton.setFont(new Font("Times New Roman", Font.BOLD, 20));
btnNewButton.setBounds(267, 41, 160, 53);
contentPane.add(btnNewButton);

JButton button = new JButton("\u0417\u0430\u0434\u0440\u0438\u0442\u0438\u0432\u0456\u0434\u0430\u0434\u0435");
button.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        dispose();
    }
});
button.setBounds(379, 163, 118, 23);
contentPane.add(button);

JButton btnNewButton_1 = new JButton("\u0417\u0430\u0434\u0438\u0438\u0441\u0438\u0442\u0438");
btnNewButton_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        try {
            String n1="", step="";
            n1=textField.getText();
            step=textField_1.getText();
            String buf=n1+" "+step;
            String file_name="D://obuchenie/2 курс 4 семестр/Алгоритми та методи
обчислень/Лабораторна робота №1/cyclic.txt";
            lab1.write(file_name, buf);

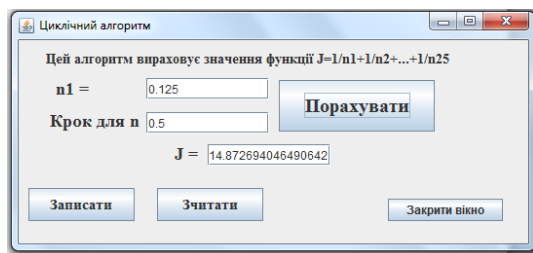
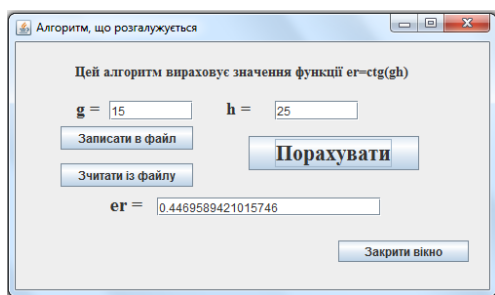
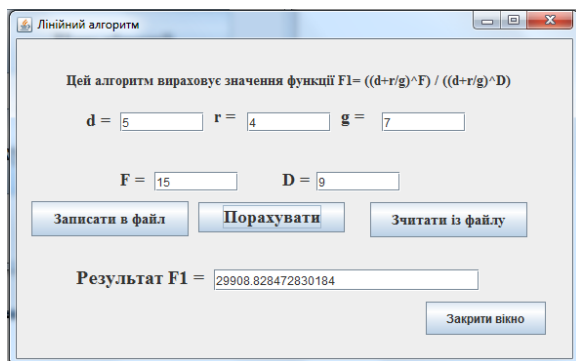
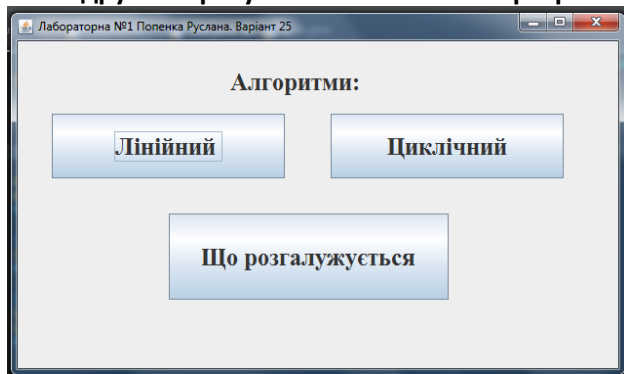
        } catch (Exception e2) {
            JOptionPane.showMessageDialog(null, "Помилка");
        }
    }
});
btnNewButton_1.setFont(new Font("Times New Roman", Font.BOLD, 15));
btnNewButton_1.setBounds(10, 152, 109, 34);
contentPane.add(btnNewButton_1);

JButton button_1 = new JButton("\u0417\u0447\u0438\u0442\u0438\u0442\u0438");
button_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String str="";
        try {
            str=lab1.read("D://obuchenie/2 курс 4 семестр/Алгоритми та методи
обчислень/Лабораторна робота №1/cyclic.txt");
        } catch (FileNotFoundException e3) {
            e3.printStackTrace();
        }
        String strVector [];
        strVector=str.split(" ");
        textField.setText(strVector[0]);
        textField_1.setText(strVector[1]);
    }
});
button_1.setFont(new Font("Times New Roman", Font.BOLD, 15));
button_1.setBounds(142, 152, 109, 34);
contentPane.add(button_1);
}

}

```

## 5. Роздруківка результатів виконання програми:



## 7. Аналіз результатів:

Виконуючи цю лабораторну роботу, я навчився створювати блок-схеми лінійного, розгалуженого та циклічного алгоритмів за допомогою редактора схем та діаграм Microsoft Office Visio. У відповідності до блок-схеми я створив програми для обчислення виразів для даних алгоритмів.

Результати роботи програм демонструють їхню працездатність та можливість перевірки ними коректності введених даних. У разі неправильного формату введених даних або невиконанні певних умов (наприклад, заборони ділення на нуль) програмою видається повідомлення про помилку.

Також введені дані можна записувати і зчитувати з файлу.