

**LAPORAN PRAKTIKUM
AGORITMA PEMROGRAMAN**

“PROGRAM OPERASI LOGIKA”

DISUSUN OLEH:
MUHAMMAD FEDORA ARGADYAKSA
2511533016

DOSEN PENGAMPU:
Dr. WAHYUDI, S.T, M.T
ASISTEN PRAKTIKUM:
RAHMAD DWIRIZKI



**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
PROGRAM STUDI INFORMATIKA
UNIVERSITAS ANDALAS**

2025

TUGAS MINGGUAN ALGORITMA DAN PEMROGRAMAN

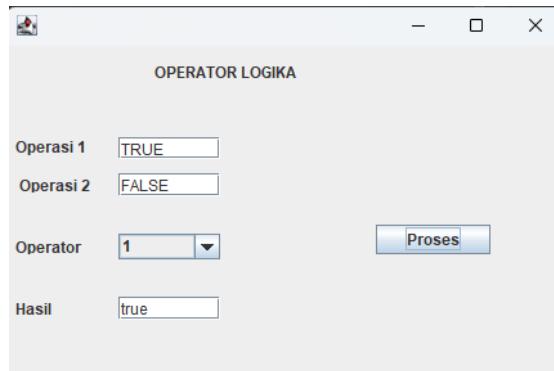
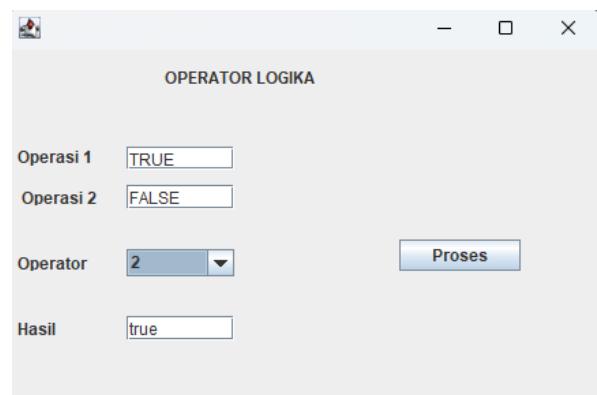
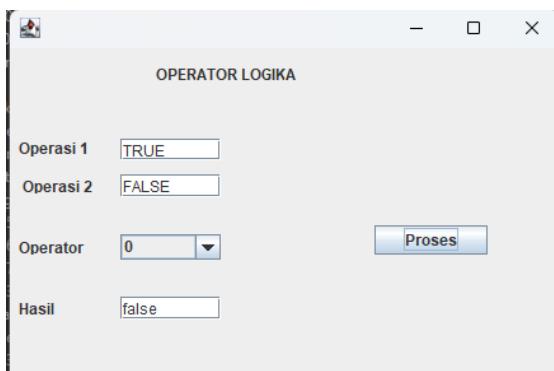
1. PROGRAM JAVA

```
1 package guipekan8_2511533016;
2
3 import java.awt.EventQueue;
4
5 public class TugasOperasiLogika_2511533016 extends JFrame {
6
7     private static final long serialVersionUID = 1L;
8     private JPanel contentPane;
9     private JTextField txtBill;
10    private JTextField txtBill2;
11    private JTextField txtHasil;
12    private pesanPeringatan;
13
14    /**
15     * Launch the application.
16     */
17    public static void main(String[] args) {
18        EventQueue.invokeLater(new Runnable() {
19            public void run() {
20                try {
21                    TugasOperasiLogika_2511533016 frame = new TugasOperasiLogika_2511533016();
22                    frame.setVisible(true);
23                } catch (Exception e) {
24                    e.printStackTrace();
25                }
26            }
27        });
28    }
29
30    /**
31     * Create the frame.
32     */
33    public TugasOperasiLogika_2511533016() {
34        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
35        setBounds(100, 100, 450, 300);
36        contentPane = new JPanel();
37        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
38        setContentPane(contentPane);
39        contentPane.setLayout(null);
40
41        JLabel lblNewLabel = new JLabel("OPERATOR LOGIKA");
42        lblNewLabel.setBounds(115, 10, 142, 20);
43        contentPane.add(lblNewLabel);
44
45        txtBill = new JTextField();
46        txtBill.setBounds(87, 71, 79, 17);
47        contentPane.add(txtBill);
48        txtBill.setColumns(10);
49
50        JComboBox comboBox = new JComboBox();
51        comboBox.setModel(new DefaultComboBoxModel(new String[]{"0", "1", "2"}));
52        comboBox.setBounds(87, 146, 79, 20);
53        contentPane.add(comboBox);
54
55        txtBill2 = new JTextField();
56        txtBill2.setBounds(87, 99, 79, 18);
57        contentPane.add(txtBill2);
58        txtBill2.setColumns(10);
59
60        JLabel lblNewLabel_1 = new JLabel("Operasi 1");
61        lblNewLabel_1.setBounds(7, 71, 70, 15);
62        contentPane.add(lblNewLabel_1);
63
64        JLabel lblNewLabel_2 = new JLabel("Operasi 2");
65        lblNewLabel_2.setBounds(10, 102, 67, 12);
66        contentPane.add(lblNewLabel_2);
67
68        JLabel lblNewLabel_3 = new JLabel("Operator");
69        lblNewLabel_3.setBounds(7, 150, 72, 12);
70        contentPane.add(lblNewLabel_3);
71
72        JLabel lblNewLabel_4 = new JLabel("Hasil");
73        lblNewLabel_4.setBounds(7, 198, 44, 12);
74        contentPane.add(lblNewLabel_4);
75
76        txtHasil = new JTextField();
77        txtHasil.setBounds(7, 236, 142, 20);
78        contentPane.add(txtHasil);
79
80        /**
81         * Add event listeners here.
82         */
83
84        /**
85         * Add action listeners here.
86         */
87
88        /**
89         * Add item listeners here.
90         */
91
92        /**
93         * Add key listeners here.
94         */
95
96        /**
97         * Add mouse listeners here.
98         */
99
100    }
101}
```

```

101         txtHasil = new JTextField();
102         txtHasil.setBounds(87, 195, 79, 18);
103         contentPane.add(txtHasil);
104         txtHasil.setColumns(10);
105
106         JButton btnNewButton = new JButton("Proses");
107         btnNewButton.addActionListener(new ActionListener() {
108             public void actionPerformed(ActionEvent e) {
109                 boolean hasil = false;
110                 try {
111                     String text1 = txtBil1.getText().trim();
112                     String text2 = txtBil2.getText().trim();
113                     if(txtBil1.getText().trim().isEmpty()) {
114                         pesanPeringatan("nilai 1 harus di isi");
115                     } else
116                     if(txtBil2.getText().trim().isEmpty()) {
117                         pesanPeringatan ("nilai 2 harus di isi");
118                     }
119                     if(!(text1.equalsIgnoreCase("true") || text1.equalsIgnoreCase("false"))) {
120                         pesanError("nilai 1 harus di isi true atau false");
121                     } else
122                     if (!(text2.equalsIgnoreCase("true") || text2.equalsIgnoreCase("false"))) {
123                         pesanError("nilai 2 harus di isi true atau false");
124                     }
125                     boolean a1 = Boolean.parseBoolean(text1);
126                     boolean a2= Boolean.parseBoolean(text2);
127                     int z = comboBox.getSelectedIndex();
128                     if (z==0) {hasil = a1 && a2;}
129                     if (z==1) {hasil = a1 || a2;}
130                     if (z==2) {hasil = !a1;}
131                 }
132             catch (Exception ex) {
133                 pesanError("nilai 1 dan 2 harus berupa true atau false");
134             }
135             txtHasil.setText(String.valueOf(hasil));
136         }
137     });
138     btnNewButton.setBounds (287, 139, 89, 23);
139     contentPane.add(btnNewButton);
140
141
142 }
143

```



2. PSEUDOCODE

Judul=

PROGRAM OPERASI LOGIKA

Deklarasi=

HASIL, A1, A2 = BOOLEAN
Z = INTEGER

ALGORITMA:

Input A1 ← true/false

Input A2 ← true/false

Input Z (0 = AND, 1 = OR, 2 = NOT)

JIKA Z ← 0

HASIL ← A1 && A2

JIKA Z ← 1

HASIL ← A1 || A2

JIKA Z ← 2

HASIL ← !A1

END PROGRAM

3. FLOWCHART

