МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**ЗВІТ**

**з лабораторної роботи № 2**

**«СТВОРЕННЯ ОБРОБНИКІВ ДЛЯ ЕЛЕМЕНТІВ УПРАВЛІННЯ У**

**ANDROID-ЗАСТОСУНКАХ. РОЗМІЩЕННЯ ВІДЖЕТІВ У МОБІЛЬНИХ ЗАСТОСУНКАХ НА ГОЛОВНОМУ ЕКРАНІ. РОБОТА ЗІ СТИЛЯМИ»**

**Виконав:**

студент 3-го курсу, групи КП-93, спеціальності 121 – Інженерія програмного забезпечення

*Інюшев Артем Владиславович*

**Перевірив:**

к. т. н, старший викладач

*Хайдуров Владислав Володимирович*

Київ – 2021

**Завдання до лабораторної роботи**

1. Ознайомитись із усіма теоретичними відомостями до лабораторної роботи.

2. Розробити функціональну частину для простого калькулятора, інтерфейс

якого був розроблений у попередній лабораторній роботі. Для кожної

кнопки створити власний обробник подій.

3. Створити копію проекту, який повністю реалізовано у пункті 2.Замінити

обробник подій для кожної кнопки одним обробником для масиву кнопок,

що мають текстові надписи 0, 1, 2, ..., 9. Обробник виділити окремим

методом (функцією). У звіт додати програмний код та результати роботи

Android-програми.

4. Створити аналогічний калькулятор, що має два текстові поля, у які

користувач самостійно вводить два числа. Для вибору операцій (+, -, \*, /) а на

головному вікні (Main\_Activity) створити Spinner. Для виконання

обрахунків додати до інтерфейсу кнопку (Button).

У звіт додати програмний код та результати роботи Android-програми.

5. За аналогією до пункту 2 створити простий проект гру хрестики-нулики

для поля розміром 4х4. Кожна клітинка повинна мати список Spinner. У

звіт додати програмний код та результати роботи Android-програми.

6. Розробити три мобільних застосунки, кожен з яких складається з одного

activity. У першому застосунку має бути представлений інтерфейс, що

складається з восьми кнопок (рис. 37). Відстань між кнопками – 2%. Перша

і остання кнопка повинні бути на рівній відстані від країв екрану.

7. Інтерфейс другого програмного додатку на рис. 38. Верстка повинна

здійснюватися з використанням RelativeLayout (не використовувати

LinearLayout).

8. Третій додаток (рис. 39) повинен у інтерфейсі кнопку, яка повинна бути

вирівняна по центру екрана. Колір контуру кнопки #505050. Товщину

контуру обрати за бажанням від 5 до 12. Радіус округлений 24dp. Колір

фону екрану #FFFFFF. При натисканні на кнопку її колір повинен

змінюватися на світло-зелений. Висота кнопки повинна складати 20% від

висоти екрана.

**Мета роботи**

Ознайомитись із основними принципами та методами

створення обробників для елементів управління в Android-застосунках.

Навчитись створювати власті стилі, ознайомитись із розміткою при створенні

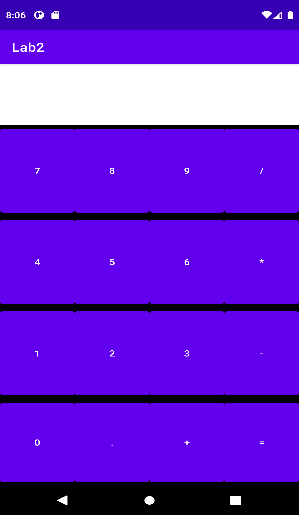
інтерфейсів в Android-застосунках.

**Перше завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:background="@color/black"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">  <TableLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">  <TableRow  android:minHeight="50dp">  <TextView  android:id="@+id/ExpressionField"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:background="@color/white"  android:textColor="@color/black"  android:textSize="30sp" />  </TableRow>  <TableRow  android:minHeight="50dp">  <TextView  android:id="@+id/ResultField"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:background="@color/white"  android:textColor="@color/black"  android:textSize="30sp" />  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:onClick="onSevenClick"  android:text="7" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="8"  android:onClick="onEightClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="9"  android:onClick="onNineClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="/"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="4"  android:onClick="onFourClick" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="5"  android:onClick="onFiveClick" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="6"  android:onClick="onSixClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="\*"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="1"  android:onClick="OnOneClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="2"  android:onClick="onTwoClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="3"  android:onClick="onThreeClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="-"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="0"  android:onClick="onZeroClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="."  android:onClick="OnOperationClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="+"  android:onClick="OnOperationClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="="  android:onClick="OnOperationClick"/>  </TableRow>  </TableLayout>  </androidx.constraintlayout.widget.ConstraintLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

|  |
| --- |
| MainActivity.java |
| package com.example.lab1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  import android.view.View;  import android.widget.Button;  import android.widget.TextView;  public class MainActivity extends AppCompatActivity {  private TextView ExpressionField;  private TextView ResultField;  private String CurrentNumber = "";  private Character CurrentOperation = ' ';  private Double CurrentResult = 0.0;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  ExpressionField = (TextView) findViewById(R.id.ExpressionField);  ResultField = (TextView) findViewById(R.id.ResultField);  }  private double ParseDouble(String value) {  try {  return Double.parseDouble(value);  }  catch (NumberFormatException e) {  ResultField.setText("Error");  // do something in case of error  }  return -100;  }  private void Calculate() throws Exception {  double result = CurrentResult;  double number = ParseDouble(CurrentNumber);  double res;  switch (CurrentOperation) {  case '+':  res = result + number;  ResultField.setText(Double.toString(res));  break;  case '-':  res = result - number;  ResultField.setText(Double.toString(res));  break;  case '\*':  res = result \* number;  ResultField.setText(Double.toString(res));  break;  case '/':  res = result / number;  ResultField.setText(Double.toString(res));  break;  default:  throw new Exception("Missing operation");  }  }  public void OnNumberClick(View view){  Button button = (Button) view;  CharSequence numberStr = button.getText().toString();  if(CurrentOperation == '=') {  ExpressionField.setText(numberStr);  CurrentOperation = ' ';  }  else {  ExpressionField.append(numberStr);  }  if(CurrentOperation == ' '){  ResultField.append(numberStr);  }  else {  CurrentNumber += numberStr;  try {  Calculate();  } catch (Exception e) {  e.getMessage();  }  }  }  public void OnOperationClick(View view){  Button button = (Button) view;  CharSequence operation = button.getText();  if(operation.toString().equals(".")){  String resultField = ExpressionField.getText().toString();  if(resultField.charAt(resultField.length() - 1) != '.'){  OnNumberClick(view);  }  return;  }  // check if user has added something after operation  if(CurrentNumber == "" && CurrentOperation != ' '){  String resultField = ExpressionField.getText().toString();  resultField = resultField.substring(0, resultField.length() - 1);  ExpressionField.setText(resultField);  }  CurrentOperation = operation.charAt(0);  ExpressionField.append(operation);  CurrentNumber = "";  if(CurrentOperation == '='){  ExpressionField.append(ResultField.getText());  ResultField.setText("");  CurrentResult = 0.0;  }  else {  CurrentResult = Double.parseDouble(ResultField.getText().toString());  }  }  public void OnOneClick(View view){  OnNumberClick(view);  }  public void onTwoClick(View view){  OnNumberClick(view);  }  public void onThreeClick(View view){  OnNumberClick(view);  }  public void onFourClick(View view){  OnNumberClick(view);  }  public void onFiveClick(View view){  OnNumberClick(view);  }  public void onSixClick(View view){  OnNumberClick(view);  }  public void onSevenClick(View view){  OnNumberClick(view);  }  public void onEightClick(View view){  OnNumberClick(view);  }  public void onNineClick(View view){  OnNumberClick(view);  }  public void onZeroClick(View view){  OnNumberClick(view);  }  } |

Зображення програми:

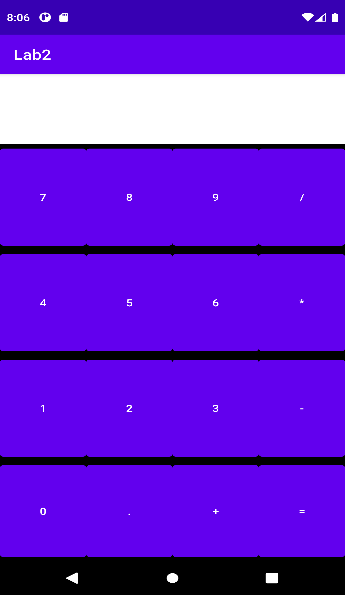


**Друге завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:background="@color/black"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">  <TableLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">  <TableRow  android:minHeight="50dp">  <TextView  android:id="@+id/ExpressionField"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:background="@color/white"  android:textColor="@color/black"  android:textSize="30sp" />  </TableRow>  <TableRow  android:minHeight="50dp">  <TextView  android:id="@+id/ResultField"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:background="@color/white"  android:textColor="@color/black"  android:textSize="30sp" />  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:onClick="OnNumberClick"  android:text="7" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="8"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="9"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="/"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="4"  android:onClick="OnNumberClick" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="5"  android:onClick="OnNumberClick" />  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="6"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="\*"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="1"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="2"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="3"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="-"  android:onClick="OnOperationClick"/>  </TableRow>  <TableRow  android:minHeight="150dp">  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="0"  android:onClick="OnNumberClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="."  android:onClick="OnOperationClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="+"  android:onClick="OnOperationClick"/>  <Button  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="="  android:onClick="OnOperationClick"/>  </TableRow>  </TableLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

|  |
| --- |
| MainActivity.java |
| package com.example.lab1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  import android.view.View;  import android.widget.Button;  import android.widget.TextView;  public class MainActivity extends AppCompatActivity {  private TextView ExpressionField;  private TextView ResultField;  private String CurrentNumber = "";  private Character CurrentOperation = ' ';  private Double CurrentResult = 0.0;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  ExpressionField = (TextView) findViewById(R.id.ExpressionField);  ResultField = (TextView) findViewById(R.id.ResultField);  }  private double ParseDouble(String value) {  try {  return Double.parseDouble(value);  }  catch (NumberFormatException e) {  ResultField.setText("Error");  // do something in case of error  }  return -100;  }  private void Calculate() throws Exception {  double result = CurrentResult;  double number = ParseDouble(CurrentNumber);  double res;  switch (CurrentOperation) {  case '+':  res = result + number;  ResultField.setText(Double.toString(res));  break;  case '-':  res = result - number;  ResultField.setText(Double.toString(res));  break;  case '\*':  res = result \* number;  ResultField.setText(Double.toString(res));  break;  case '/':  res = result / number;  ResultField.setText(Double.toString(res));  break;  default:  throw new Exception("Missing operation");  }  }  public void OnNumberClick(View view){  Button button = (Button) view;  CharSequence numberStr = button.getText().toString();  if(CurrentOperation == '=') {  ExpressionField.setText(numberStr);  CurrentOperation = ' ';  }  else {  ExpressionField.append(numberStr);  }  if(CurrentOperation == ' '){  ResultField.append(numberStr);  }  else {  CurrentNumber += numberStr;  try {  Calculate();  } catch (Exception e) {  e.getMessage();  }  }  }  public void OnOperationClick(View view){  Button button = (Button) view;  CharSequence operation = button.getText();  if(operation.toString().equals(".")){  String resultField = ExpressionField.getText().toString();  if(resultField.charAt(resultField.length() - 1) != '.'){  OnNumberClick(view);  }  return;  }  // check if user has added something after operation  if(CurrentNumber == "" && CurrentOperation != ' '){  String resultField = ExpressionField.getText().toString();  resultField = resultField.substring(0, resultField.length() - 1);  ExpressionField.setText(resultField);  }  CurrentOperation = operation.charAt(0);  ExpressionField.append(operation);  CurrentNumber = "";  if(CurrentOperation == '='){  ExpressionField.append(ResultField.getText());  ResultField.setText("");  CurrentResult = 0.0;  }  else {  CurrentResult = Double.parseDouble(ResultField.getText().toString());  }  }  } |

Зображення програми:

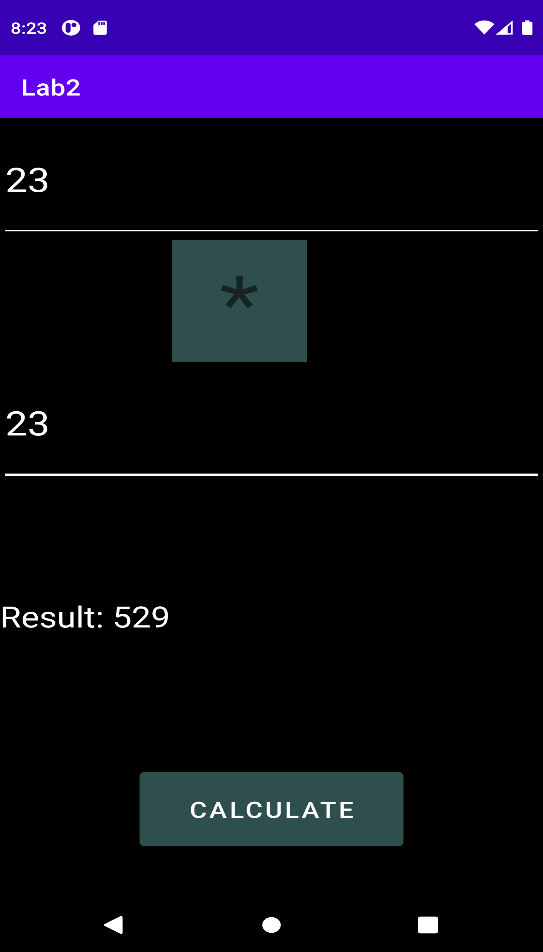


**Третє завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:background="@color/black">  <EditText  android:id="@+id/FirstNumber"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:backgroundTint="@color/white"  android:textColorHint="@color/grey"  android:textColor="@color/white"  android:hint="First number"  android:inputType="number"  android:textSize="30sp"/>  <Spinner  android:id="@+id/Spinner"  android:layout\_width="150dp"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_gravity="center"/>  <EditText  android:id="@+id/SecondNumber"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:backgroundTint="@color/white"  android:textColorHint="@color/grey"  android:textColor="@color/white"  android:hint="Second number"  android:inputType="number"  android:textSize="30sp"/>  <TextView  android:id="@+id/Result"  android:layout\_width="match\_parent"  android:layout\_height="150dp"  android:layout\_marginTop="100sp"  android:textSize="26sp"  android:textColor="@color/white"/>  <Button  android:layout\_width="200dp"  android:layout\_height="78dp"  android:layout\_gravity="center"  android:layout\_marginBottom="40sp"  android:backgroundTint="@color/darkGrey"  android:text="Calculate"  android:textSize="20sp"  android:onClick="Calculate"/>  </LinearLayout> |

|  |
| --- |
| MainActivity.java |
| package com.example.lab1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  import android.widget.ArrayAdapter;  import android.widget.EditText;  import android.widget.Spinner;  import android.widget.TextView;  import android.view.View;  public class MainActivity extends AppCompatActivity {  private Spinner OperationSpin;  private EditText FirstNumber;  private EditText SecondNumber;  private TextView ResultView;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  FirstNumber = (EditText) findViewById(R.id.FirstNumber);  SecondNumber = (EditText) findViewById(R.id.SecondNumber);  ResultView = (TextView) findViewById(R.id.Result);  OperationSpin = (Spinner) findViewById(R.id.Spinner);  ArrayAdapter adapter = ArrayAdapter.createFromResource(this, R.array.operations, R.layout.spinner\_item);  adapter.setDropDownViewResource(R.layout.spinner\_item);  OperationSpin.setAdapter(adapter);  }  public void Calculate(View view){  String firstNumStr = FirstNumber.getText().toString();  if(firstNumStr.isEmpty()){  ResultView.setText("Error - first number is empty");  return;  }  String secondNumStr = SecondNumber.getText().toString();  if(secondNumStr.isEmpty()){  ResultView.setText("Error - second number is empty");  return;  }  String operation = OperationSpin.getSelectedItem().toString();  int firstNum = Integer.parseInt(firstNumStr);  int secondNum = Integer.parseInt(secondNumStr);  String result = "Result: ";  switch (operation){  case "+":  result += Integer.toString(firstNum + secondNum);  break;  case "-":  result += Integer.toString(firstNum - secondNum);  break;  case "\*":  result += Integer.toString(firstNum \* secondNum);  break;  case "/":  if(secondNum == 0) {  ResultView.setText("Error - division by Zero");  return;  }  result += Double.toString(new Double(firstNum) / secondNum);  break;  default:  result += "Error!!!";  break;  }  ResultView.setText(result);  }  } |

Зображення програми:

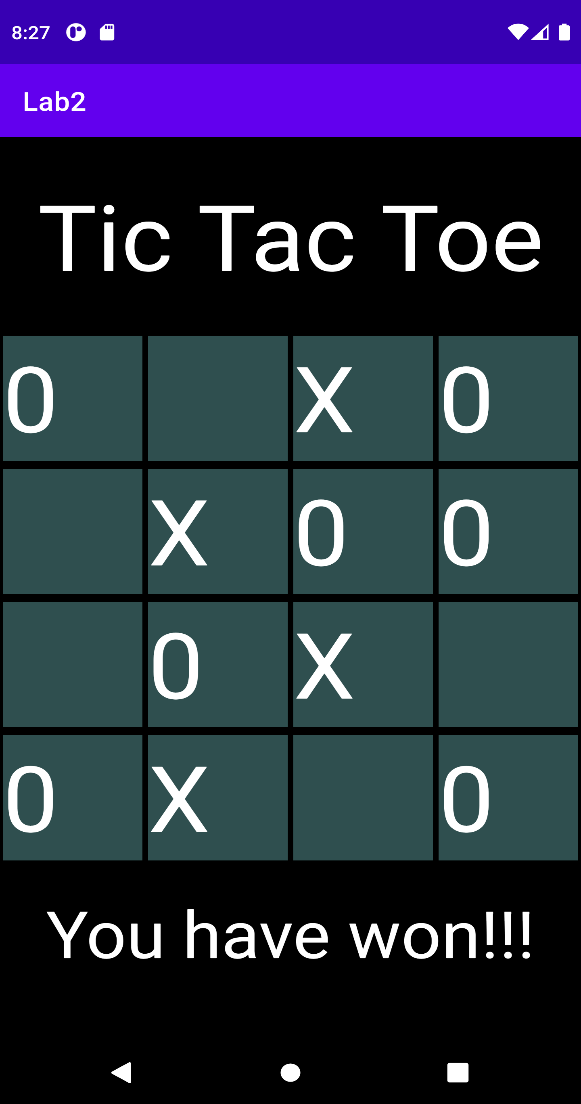


**Четверте завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/black"  tools:context=".MainActivity">  <TextView  android:id="@+id/Header"  android:layout\_width="match\_parent"  android:layout\_height="150dp"  android:gravity="center"  android:text="Tic Tac Toe"  android:textColor="@color/white"  android:textSize="70dp"  app:layout\_constraintTop\_toTopOf="parent"/>  <TableLayout  android:id="@+id/Table"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  app:layout\_constraintTop\_toBottomOf="@id/Header">  <TableRow  android:minHeight="100dp">  <Spinner  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  </TableRow>  <TableRow  android:minHeight="100dp"  android:layout\_marginTop="2sp">  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  </TableRow>  <TableRow  android:minHeight="100dp"  android:layout\_marginTop="2sp">  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  </TableRow>  <TableRow  android:minHeight="100dp"  android:layout\_marginTop="2sp">  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  <Spinner  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:background="@color/darkGrey"/>  </TableRow>  </TableLayout>  <TextView  android:id="@+id/Result"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="20sp"  android:gravity="center"  android:textSize="50dp"  android:textColor="@color/white"  app:layout\_constraintTop\_toBottomOf="@+id/Table"/>  </androidx.constraintlayout.widget.ConstraintLayout> |

|  |
| --- |
| MainActivity.java |
| package com.example.lab1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  import android.widget.AdapterView;  import android.widget.TableLayout;  import android.widget.TableRow;  import android.widget.Spinner;  import android.widget.ArrayAdapter;  import android.view.View;  import android.widget.TextView;  public class MainActivity extends AppCompatActivity {  private TextView ResultView;  private Spinner [][] Spinners = new Spinner[4][4];  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  ResultView = (TextView) findViewById(R.id.Result);  TableLayout table = (TableLayout) findViewById(R.id.Table);  int rowsCount = table.getChildCount();  for(int i = 0; i < rowsCount; i++){  TableRow row = (TableRow) table.getChildAt(i);  int spinnersCount = row.getChildCount();  for(int j = 0; j < spinnersCount; j++){  Spinner spinner = (Spinner) row.getChildAt(j);  ArrayAdapter adapter = ArrayAdapter.createFromResource(this, R.array.options, R.layout.spinner\_item);  adapter.setDropDownViewResource(R.layout.spinner\_dropdown\_item);  spinner.setAdapter(adapter);  spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {  @Override  public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {  if(CheckIfGameFinished()) {  ResultView.setText("You have won!!!");  }  else {  ResultView.setText("");  }  }  @Override  public void onNothingSelected(AdapterView<?> adapterView) { }  });  Spinners[i][j] = spinner;  }  }  }  private boolean CheckIfGameFinished(){  boolean diagonals = CheckDiagonals();  boolean rows\_cols = CheckRowsAndCols();  if(diagonals || rows\_cols){  return true;  }  return false;  }  private boolean CheckDiagonals(){  boolean finishedDiagonal = true;  boolean finishedDiagonalReverse = true;  String diagonalFirst = Spinners[0][0].getSelectedItem().toString();  String diagonalReverseFirst = Spinners[0][3].getSelectedItem().toString();  if(diagonalFirst.equals("")){  finishedDiagonal = false;  }  if(diagonalReverseFirst.equals("")){  finishedDiagonalReverse = false;  }  for (int i = 1; i < 4; i++) {  if (!diagonalFirst.equals(Spinners[i][i].getSelectedItem().toString())) {  finishedDiagonal = false;  }  if (!diagonalReverseFirst.equals(Spinners[i][3 - i].getSelectedItem().toString())) {  finishedDiagonalReverse = false;  }  if (!finishedDiagonal && !finishedDiagonalReverse) {  break;  }  }  if (finishedDiagonal || finishedDiagonalReverse) {  return true;  }  return false;  }  private boolean CheckRowsAndCols(){  for (int i = 0; i < 4; i++) {  boolean finishedRow = true;  boolean finishedCol = true;  String rowFirst = Spinners[0][i].getSelectedItem().toString();  String colFirst = Spinners[i][0].getSelectedItem().toString();  if(rowFirst.equals("")){  finishedRow = false;  }  if(colFirst.equals("")){  finishedCol = false;  }  for (int j = 1; j < 4; j++) {  if(!rowFirst.equals(Spinners[j][i].getSelectedItem().toString())) {  finishedRow = false;  }  if (!colFirst.equals(Spinners[i][j].getSelectedItem().toString())) {  finishedCol = false;  }  if(!finishedRow && !finishedCol) {  break;  }  }  if(finishedRow || finishedCol) {  return true;  }  }  return false;  }  } |

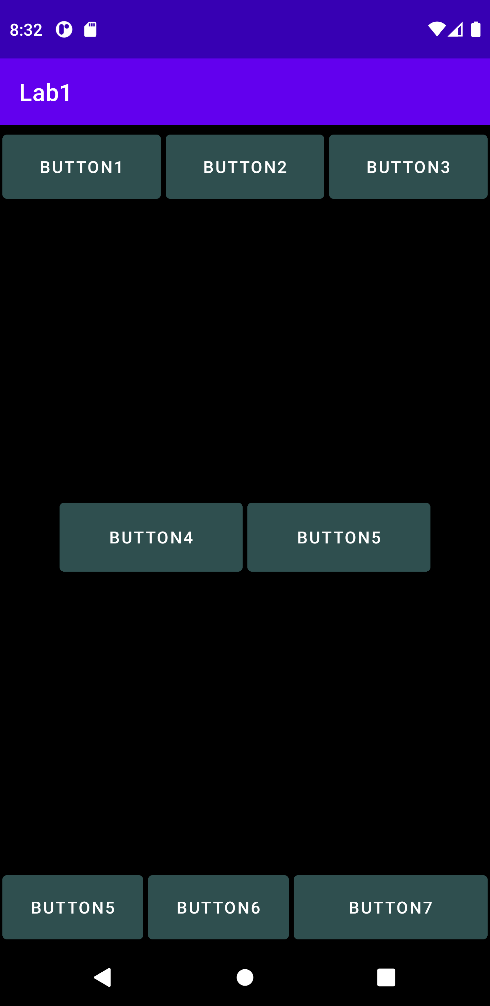
Зображення програми:



**П’яте завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity"  android:background="@color/black">  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="70sp"  app:layout\_constraintTop\_toTopOf="parent">  <Button  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:text="Button1"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:text="Button2"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:text="Button3"  android:backgroundTint="@color/darkGrey"/>  </LinearLayout>  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="70sp"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintBottom\_toBottomOf="parent">  <Button  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_marginLeft="50sp"  android:layout\_marginRight="2sp"  android:text="Button4"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_marginLeft="2sp"  android:layout\_marginRight="50sp"  android:text="Button5"  android:backgroundTint="@color/darkGrey"/>  </LinearLayout>  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="70sp"  app:layout\_constraintBottom\_toBottomOf="parent">  <Button  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:text="Button5"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="1"  android:layout\_margin="2sp"  android:text="Button6"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_weight="4"  android:layout\_margin="2sp"  android:text="Button7"  android:backgroundTint="@color/darkGrey"/>  </LinearLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

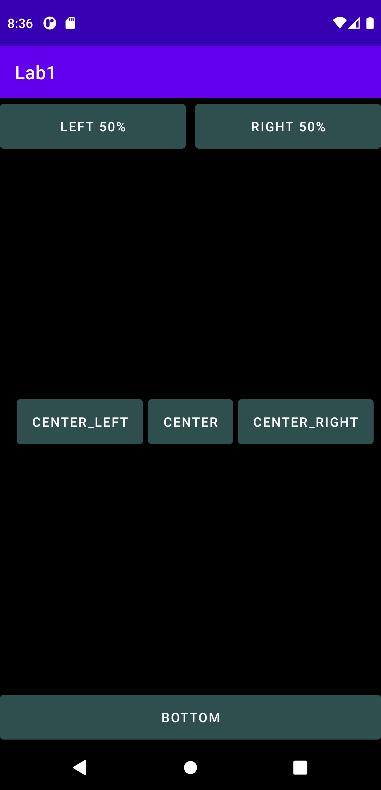
Зображення програми:



**Шосте завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <RelativeLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/black">  <View android:id="@+id/Separator"  android:layout\_width="0dp"  android:layout\_height="0dp"  android:layout\_centerHorizontal="true"/>  <Button  android:layout\_width="0sp"  android:layout\_height="60sp"  android:layout\_alignRight="@id/Separator"  android:layout\_alignParentLeft="true"  android:layout\_marginRight="5sp"  android:text="Left 50%"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="0sp"  android:layout\_height="60sp"  android:layout\_alignLeft="@id/Separator"  android:layout\_alignParentRight="true"  android:layout\_marginLeft="5sp"  android:text="Right 50%"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="60sp"  android:layout\_centerVertical="true"  android:layout\_toLeftOf="@+id/Center"  android:text="Center\_left"  android:layout\_marginRight="5sp"  android:backgroundTint="@color/darkGrey"/>  <Button  android:id="@+id/Center"  android:layout\_width="wrap\_content"  android:layout\_height="60sp"  android:layout\_centerVertical="true"  android:layout\_centerHorizontal="true"  android:text="Center"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="60sp"  android:layout\_centerVertical="true"  android:layout\_toRightOf="@+id/Center"  android:text="Center\_right"  android:layout\_marginLeft="5sp"  android:backgroundTint="@color/darkGrey"/>  <Button  android:layout\_width="match\_parent"  android:layout\_height="60sp"  android:layout\_centerHorizontal="true"  android:layout\_alignParentBottom="true"  android:text="Bottom"  android:backgroundTint="@color/darkGrey"/></RelativeLayout> |

Зображення програми:



**Сьоме завдання**

|  |
| --- |
| activity\_main.xml |
| <?xml version="1.0" encoding="utf-8"?>  <RelativeLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/white">  <Button  android:layout\_width="240sp"  android:layout\_height="120sp"  android:layout\_centerInParent="true"  android:background="@drawable/my\_button"  android:textSize="30dp"  android:text="Button"  android:onClick="ChangeColor"/>  </RelativeLayout> |

|  |
| --- |
| MainActivity.java |
| package com.example.lab1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  import android.view.View;  import android.widget.Button;  public class MainActivity extends AppCompatActivity {  private boolean changed = false;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  }  public void ChangeColor(View view){  Button button = (Button) view;  if(changed){  button.setBackgroundResource(R.drawable.my\_button);  changed = false;  }  else{  button.setBackgroundResource(R.drawable.my\_button\_clicked);  changed = true;  }  }  } |

Зображення програми:

**Висновок**

У процесі виконання даної лабораторної нам вдалося:

1. Розробити функціональну частину для простого калькулятора, інтерфейс

якого був розроблений у попередній лабораторній роботі. Для кожної

кнопки створити власний обробник подій.

2. Створити копію проекту, який повністю реалізовано у пункті 2.Замінити

обробник подій для кожної кнопки одним обробником для масиву кнопок,

що мають текстові надписи 0, 1, 2, ..., 9. Обробник виділити окремим

методом (функцією). У звіт додати програмний код та результати роботи

Android-програми.

3. Створити аналогічний калькулятор, що має два текстові поля, у які

користувач самостійно вводить два числа. Для вибору операцій (+, -, \*, /) а на

головному вікні (Main\_Activity) створити Spinner. Для виконання

обрахунків додати до інтерфейсу кнопку (Button).

У звіт додати програмний код та результати роботи Android-програми.

4. За аналогією до пункту 2 створити простий проект гру хрестики-нулики

для поля розміром 4х4. Кожна клітинка повинна мати список Spinner. У

звіт додати програмний код та результати роботи Android-програми.

5. Розробити три мобільних застосунки, кожен з яких складається з одного

activity. У першому застосунку має бути представлений інтерфейс, що

складається з восьми кнопок (рис. 37). Відстань між кнопками – 2%. Перша

і остання кнопка повинні бути на рівній відстані від країв екрану.

6. Інтерфейс другого програмного додатку на рис. 38. Верстка повинна

здійснюватися з використанням RelativeLayout (не використовувати

LinearLayout).

7. Третій додаток (рис. 39) повинен у інтерфейсі кнопку, яка повинна бути

вирівняна по центру екрана. Колір контуру кнопки #505050. Товщину

контуру обрати за бажанням від 5 до 12. Радіус округлений 24dp. Колір

фону екрану #FFFFFF. При натисканні на кнопку її колір повинен

змінюватися на світло-зелений. Висота кнопки повинна складати 20% від

висоти екрана.