**Міністерство освіти і науки України**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

**Лабораторна робота №1**

з дисципліни

«Об’єктно орієнтоване програмування»

на тему

“Знайомство із середовищем розробки

програм Microsoft Visual Studio та складання модульних

проєктів програм на C++”

Виконав: Перевірив:

Студент групи ІМ-22 Порєв В.М.

Кушнір Микола Миколайович

номер у списку групи: 13

Київ 2023

**Мета**

Отримати перші навички створення програм для Windows на

основі проєктів для Visual C++ з використанням Windows API і навчитися

модульному програмуванню на C++

**Завдання**

**1.** Створити у середовищі MS Visual Studio C++ проєкт з ім’ям **Lab1**.

**2.** Написати вихідний текст програми згідно варіанту завдання.

**3.** Скомпілювати вихідний текст і отримати виконуваний файл

програми.

**4.** Перевірити роботу програми. Налагодити програму.

**5.** Проаналізувати та прокоментувати результати та вихідний текст

програми.

**Варіанти:**

**B1 = 1**: Вікно діалогу з повзуном горизонтального скролінгу (Horizontal scroll Bar) та дві кнопки: [Так] і [Відміна]. Рухаючи повзунок скролінгу користувач вводить число у діапазоні від 1 до 100. Після натискування кнопки [Так] вибране число буде відображатися у головному вікні.

**B2 = 2**: Два вікна діалогу. Спочатку з’являється перше, яке має дві кнопки: [Далі >] і [Відміна]. Якщо натиснути кнопку [Далі >], то воно закриється і з’явиться друге діалогове вікно, яке має кнопки: [< Назад], [Так] і [Відміна]. Якщо натиснути кнопку [<Назад], вікно закриється і відбудеться перехід до першого вікна.

**Вихідні тексти файлів програми**

**Lab1.kt**

package com.oop.lab1  
  
*// Імпорт пакетів системи Android*import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Gravity  
import android.view.Menu  
import android.view.MenuItem  
import android.widget.TextView  
import android.widget.Toast  
import androidx.core.content.ContextCompat  
*// Імпорт пакетів модуля 1*import com.oop.lab1.module1.Module1  
import com.oop.lab1.module1.Module1Interface  
*// Імпорт пакетів модуля 2*import com.oop.lab1.module2.Module2  
import com.oop.lab1.module2.Module2Interface  
*// Імпорт пакетів модуля 3*import com.oop.lab1.module3.Module3  
import com.oop.lab1.module3.Module3Interface  
*// Оголошення констант для позначення подій натискання кнопок  
// BTN\_CANCEL = 0*private const val *BTN\_CONFIRM* = 1  
private const val *BTN\_THEN* = 2  
private const val *BTN\_BACK* = 3  
class Lab1 : AppCompatActivity(),  
 Module1Handler,  
 Module2Handler,  
 Module3Handler {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_lab1*)  
 }  
 override fun onCreateOptionsMenu(menu: Menu): Boolean {  
 val inflater = *menuInflater* inflater.inflate(R.menu.*menu\_lab1*, menu)  
 return true  
 }  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 return when (item.*itemId*) {  
 R.id.*btn\_work1\_menu\_lab1* -> {  
 runModule1()  
 true  
 }  
 R.id.*btn\_work2\_menu\_lab1* -> {  
 runModule2()  
 true  
 }  
 else -> {  
 super.onOptionsItemSelected(item)  
 }  
 }  
 }  
 private fun runModule1() {  
 val module1: Module1Interface = Module1()  
 module1.run(this, *supportFragmentManager*, this)  
 }  
 private fun runModule2() {  
 val module2: Module2Interface = Module2()  
 module2.run(this, *supportFragmentManager*, this)  
 }  
 private fun runModule3() {  
 val module3: Module3Interface = Module3()  
 module3.run(this, *supportFragmentManager*, this)  
 }  
 override fun handleModule1Result(pressedButton: Int, result: Int?) {  
 if (pressedButton == *BTN\_CONFIRM*) {  
 val textView = findViewById<TextView>(R.id.*selected\_number\_lab1*)  
 if (result != null) {  
 textView.*text* = "$result"  
 textView.*textSize* = 50f  
 val textColor = ContextCompat.getColor(this, R.color.*black*)  
 textView.setTextColor(textColor)  
 } else {  
 textView.*text* = "На жаль, ви не вибрали значення, спробуйте ще"  
 textView.*textSize* = 20f  
 val textColor = ContextCompat.getColor(this, R.color.*red*)  
 textView.setTextColor(textColor)  
 textView.*gravity* = Gravity.*CENTER* }  
 }  
 }  
 override fun handleModule2Result(pressedButton: Int) {  
 if (pressedButton == *BTN\_THEN*) {  
 runModule3()  
 }  
 }  
 override fun handleModule3Result(pressedButton: Int) {  
 when (pressedButton) {  
 *BTN\_BACK* -> {  
 runModule2()  
 }  
 *BTN\_CONFIRM* -> {  
 val text = "Ви натиснули кнопку \"Так\""  
 val duration = Toast.*LENGTH\_SHORT* Toast.makeText(this, text, duration).show()  
 }  
 }  
 }  
}

**Module1Handler.kt**

package com.oop.lab1  
  
interface Module1Handler {  
 fun handleModule1Result(pressedButton: Int, result: Int?)  
}

**Module2Handler.kt**

package com.oop.lab1  
  
interface Module2Handler {  
 fun handleModule2Result(pressedButton: Int)  
}

**Module3Handler.kt**

package com.oop.lab1  
  
interface Module3Handler {  
 fun handleModule3Result(pressedButton: Int)  
}

**Module1.kt**

package com.oop.lab1.module1  
  
*// Імпорт пакетів системи Android*import android.app.AlertDialog  
import android.app.Dialog  
import android.content.Context  
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import android.widget.SeekBar  
import android.widget.SeekBar.OnSeekBarChangeListener  
import android.widget.TextView  
import androidx.core.content.ContextCompat  
import androidx.fragment.app.DialogFragment  
import androidx.fragment.app.FragmentManager  
*// Імпорт ресурсів*import com.oop.lab1.R  
*// Імпорт інтерфейсу класу, який має callback-фунцію,  
// що викликатиметься після закриття ділогового вікна та  
// опрацьовуватиме результат, отриманий після виконання модуля*import com.oop.lab1.Module1Handler  
*// Оголошення констант для позначення подій натискання кнопок*private const val *BTN\_CANCEL* = 0  
private const val *BTN\_CONFIRM* = 1  
class Module1 : DialogFragment(), Module1Interface {  
 *// Змінна, через яку передамо результат,  
 // отриманий у діалоговому вікні, до головного модуля* private var result: Int? = null  
 *// Змінна, у яку присвоїться контекст головної активності* private lateinit var activityContext: Context  
 *// Змінна, що зберігатиме вказівник на інтерфейс класу, який має callback-фунцію* private lateinit var resultHandler: Module1Handler  
 override fun onCreateDialog(savedInstanceState: Bundle?): Dialog {  
 val builder = AlertDialog.Builder(activityContext)  
 builder.setTitle(R.string.*dialog\_name\_module1*)  
 val inflater = *layoutInflater* val view: View = inflater.inflate(R.layout.*dialog\_module1*, null)  
 builder.setView(view)  
 val textView = view.findViewById<TextView>(R.id.*current\_number\_module1*)  
 val seekBar = view.findViewById<SeekBar>(R.id.*seek\_bar\_module1*)  
 seekBar.setOnSeekBarChangeListener(object : OnSeekBarChangeListener {  
 override fun onProgressChanged(seekBar: SeekBar, progress: Int, fromUser: Boolean) {  
 result = progress + 1  
 textView.*text* = "$result"  
 }  
 override fun onStartTrackingTouch(seekBar: SeekBar) {  
 if (result == null) {  
 result = 1  
 }  
 textView.*text* = "$result"  
 textView.*textSize* = 30f  
 val textColor = ContextCompat.getColor(activityContext, R.color.*black*)  
 textView.setTextColor(textColor)  
 }  
 override fun onStopTrackingTouch(seekBar: SeekBar) {  
 textView.*textSize* = 20f  
 }  
 })  
 val buttonConfirm = view.findViewById<Button>(R.id.*btn\_confirm\_module1*)  
 buttonConfirm.setOnClickListener **{** resultHandler.handleModule1Result(*BTN\_CONFIRM*, result)  
 dismiss()  
 **}** val buttonCancel = view.findViewById<Button>(R.id.*btn\_cancel\_module1*)  
 buttonCancel.setOnClickListener **{** resultHandler.handleModule1Result(*BTN\_CANCEL*, result)  
 dismiss()  
 **}** return builder.create()  
 }  
 *// Інтерфейсний метод, за допомогою якого головна активність взаємодіятиме з модулем* override fun run(context: Context, manager: FragmentManager, handler: Module1Handler) {  
 val module1 = Module1()  
 module1.activityContext = context  
 module1.resultHandler = handler  
 module1.show(manager, "dialog\_module1")  
 }  
}

**Module1Interface.kt**

package com.oop.lab1.module1  
  
import android.content.Context  
import androidx.fragment.app.FragmentManager  
import com.oop.lab1.Module1Handler  
interface Module1Interface {  
 fun run(  
 context: Context,  
 manager: FragmentManager,  
 handler: Module1Handler  
 )  
}

**Module2.kt**

package com.oop.lab1.module2  
  
*// Імпорт пакетів системи Android*import android.app.AlertDialog  
import android.app.Dialog  
import android.content.Context  
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import androidx.fragment.app.DialogFragment  
import androidx.fragment.app.FragmentManager  
*// Імпорт ресурсів*import com.oop.lab1.R  
*// Імпорт інтерфейсу класу, який має callback-фунцію,  
// що викликатиметься після закриття ділогового вікна та  
// опрацьовуватиме результат, отриманий після виконання модуля*import com.oop.lab1.Module2Handler  
*// Оголошення констант для позначення подій натискання кнопок*private const val *BTN\_CANCEL* = 0  
private const val *BTN\_THEN* = 2  
class Module2 : DialogFragment(), Module2Interface {  
 *// Змінна, у яку присвоїться контекст головної активності* private lateinit var activityContext: Context  
 *// Змінна, що зберігатиме вказівник на інтерфейс класу, який має callback-фунцію* private lateinit var resultHandler: Module2Handler  
 override fun onCreateDialog(savedInstanceState: Bundle?): Dialog {  
 val builder = AlertDialog.Builder(activityContext)  
 builder.setTitle(R.string.*dialog\_name\_module2*)  
 val inflater = *layoutInflater* val view: View = inflater.inflate(R.layout.*dialog\_module2*, null)  
 builder.setView(view)  
 val buttonThen = view.findViewById<Button>(R.id.*btn\_then\_module2*)  
 buttonThen.setOnClickListener **{** resultHandler.handleModule2Result(*BTN\_THEN*)  
 dismiss()  
 **}** val buttonCancel = view.findViewById<Button>(R.id.*btn\_cancel\_module2*)  
 buttonCancel.setOnClickListener **{** resultHandler.handleModule2Result(*BTN\_CANCEL*)  
 dismiss()  
 **}** return builder.create()  
 }  
 *// Інтерфейсний метод, за допомогою якого головна активність взаємодіятиме з модулем* override fun run(context: Context, manager: FragmentManager, handler: Module2Handler) {  
 val module2 = Module2()  
 module2.activityContext = context  
 module2.resultHandler = handler  
 module2.show(manager, "dialog\_module2")  
 }  
}

**Module2Interface.kt**

package com.oop.lab1.module2  
  
import android.content.Context  
import androidx.fragment.app.FragmentManager  
import com.oop.lab1.Module2Handler  
interface Module2Interface {  
 fun run(  
 context: Context,  
 manager: FragmentManager,  
 handler: Module2Handler  
 )  
}

**Module3.kt**

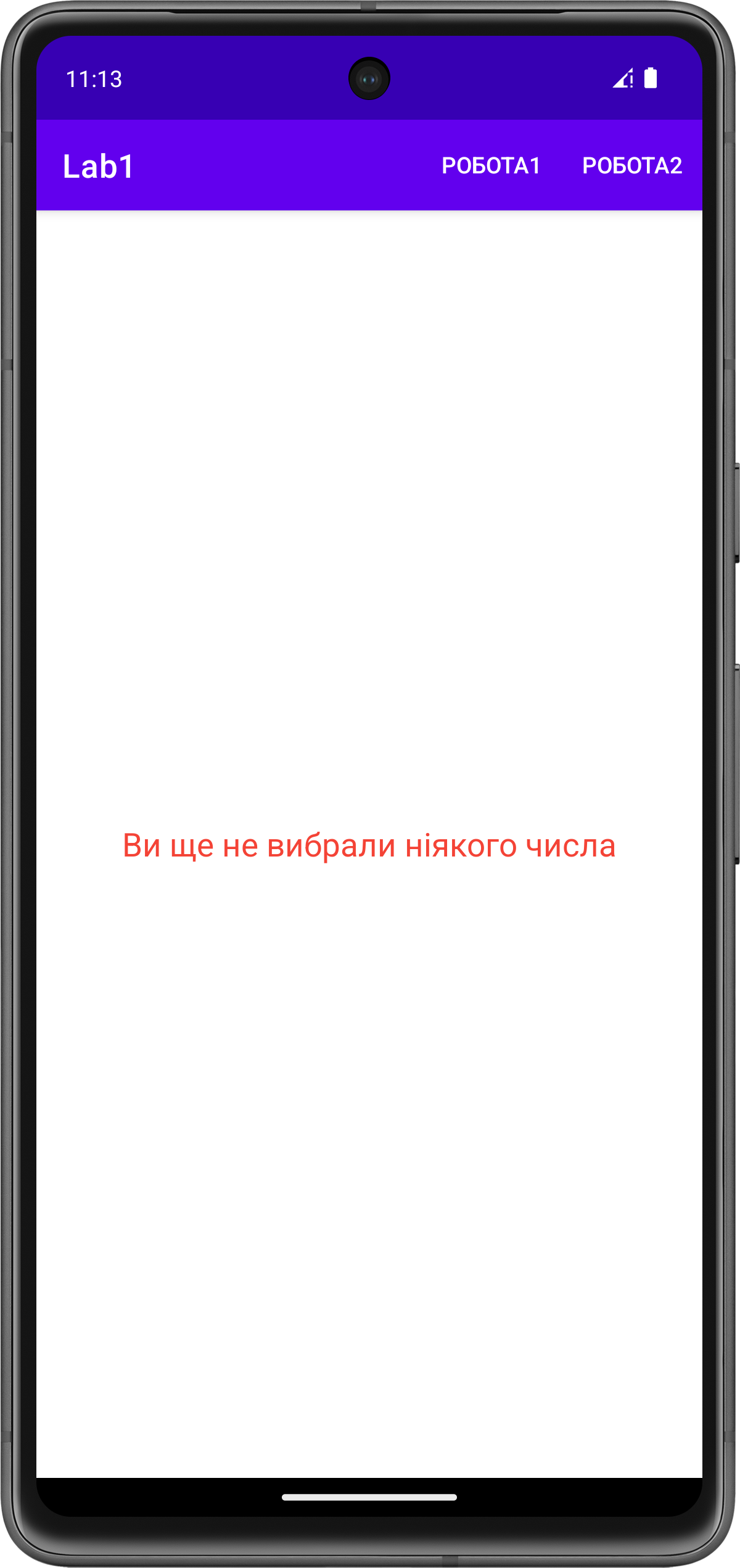
package com.oop.lab1.module3  
  
*// Імпорт пакетів системи Android*import android.app.AlertDialog  
import android.app.Dialog  
import android.content.Context  
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import androidx.fragment.app.DialogFragment  
import androidx.fragment.app.FragmentManager  
*// Імпорт ресурсів*import com.oop.lab1.R  
*// Імпорт інтерфейсу класу, який має callback-фунцію,  
// що викликатиметься після закриття ділогового вікна та  
// опрацьовуватиме результат, отриманий після виконання модуля*import com.oop.lab1.Module3Handler  
*// Оголошення констант для позначення подій натискання кнопок*private const val *BTN\_CANCEL* = 0  
private const val *BTN\_CONFIRM* = 1  
private const val *BTN\_BACK* = 3  
class Module3 : DialogFragment(), Module3Interface {  
 *// Змінна, у яку присвоїться контекст головної активності* private lateinit var activityContext: Context  
 *// Змінна, що зберігатиме вказівник на інтерфейс класу, який має callback-фунцію* private lateinit var resultListener: Module3Handler  
 override fun onCreateDialog(savedInstanceState: Bundle?): Dialog {  
 val builder = AlertDialog.Builder(activityContext)  
 builder.setTitle(R.string.*dialog\_name\_module3*)  
 val inflater = *layoutInflater* val view: View = inflater.inflate(R.layout.*dialog\_module3*, null)  
 builder.setView(view)  
 val buttonBack = view.findViewById<Button>(R.id.*btn\_back\_module3*)  
 buttonBack.setOnClickListener **{** resultListener.handleModule3Result(*BTN\_BACK*)  
 dismiss()  
 **}** val buttonConfirm = view.findViewById<Button>(R.id.*btn\_confirm\_module3*)  
 buttonConfirm.setOnClickListener **{** resultListener.handleModule3Result(*BTN\_CONFIRM*)  
 **}** val buttonCancel = view.findViewById<Button>(R.id.*btn\_cancel\_module3*)  
 buttonCancel.setOnClickListener **{** resultListener.handleModule3Result(*BTN\_CANCEL*)  
 dismiss()  
 **}** return builder.create()  
 }  
 *// Інтерфейсний метод, за допомогою якого головна активність взаємодіятиме з модулем* override fun run(context: Context, manager: FragmentManager, handler: Module3Handler) {  
 val module3 = Module3()  
 module3.activityContext = context  
 module3.resultListener = handler  
 module3.show(manager, "dialog\_module2")  
 }  
}

**Module3Interface.kt**

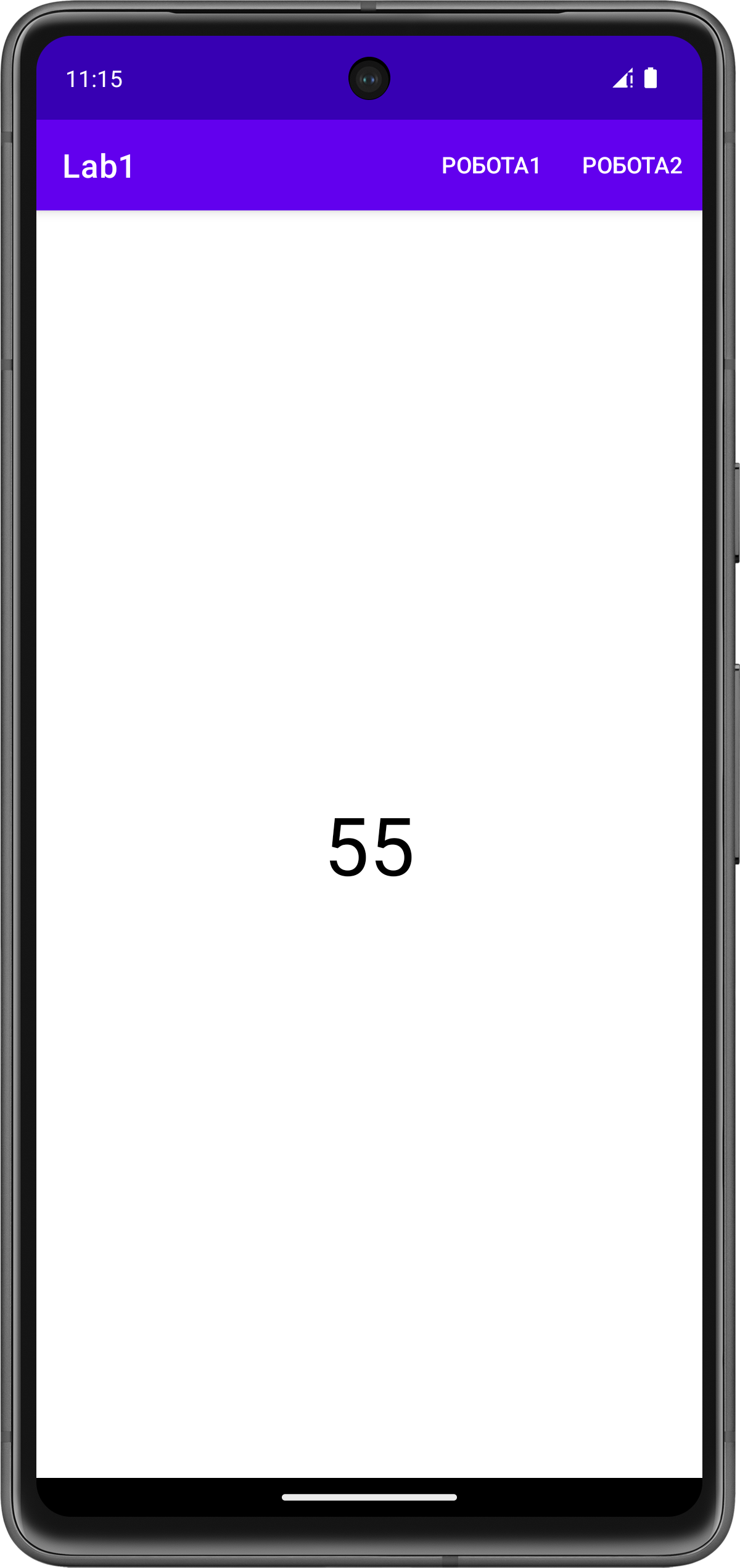
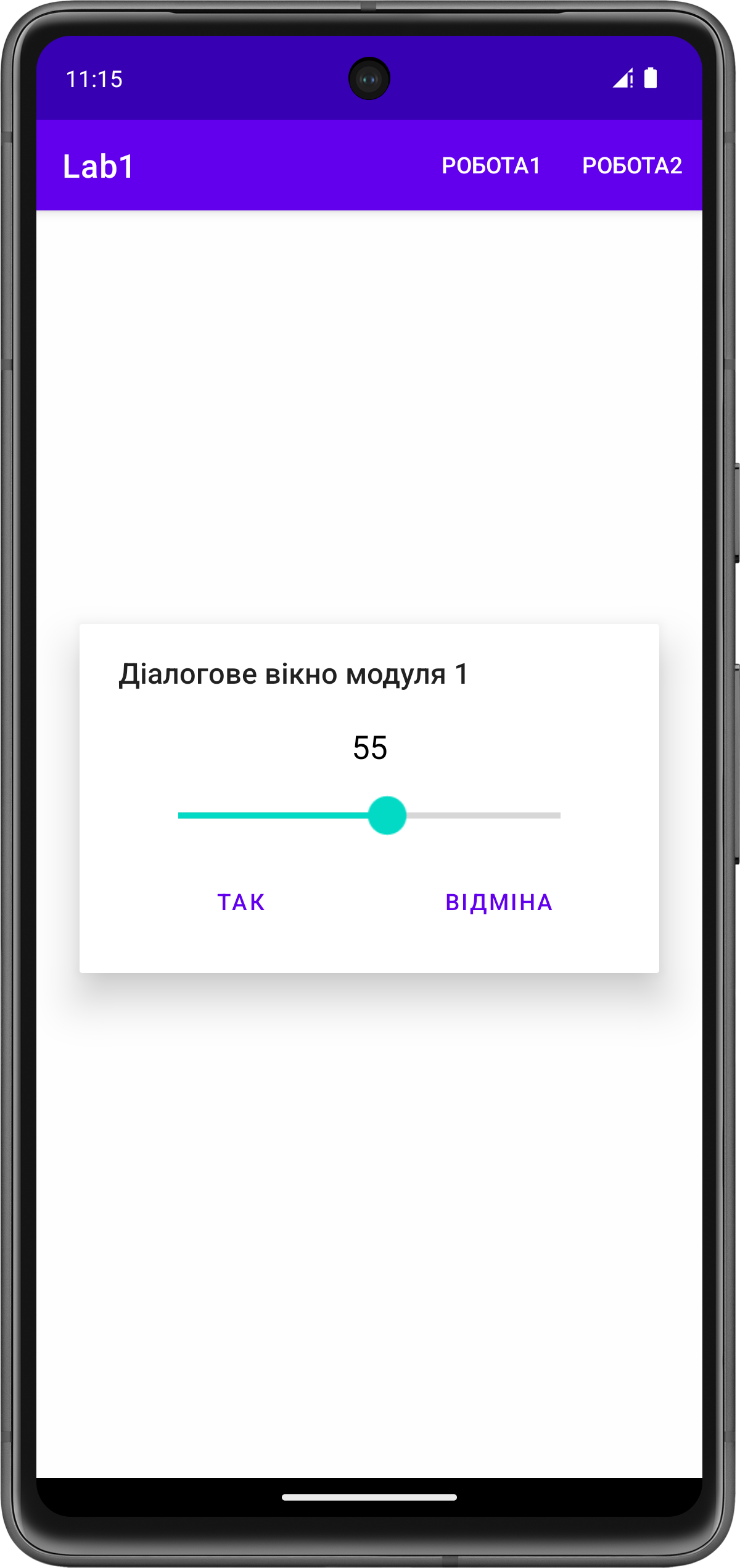
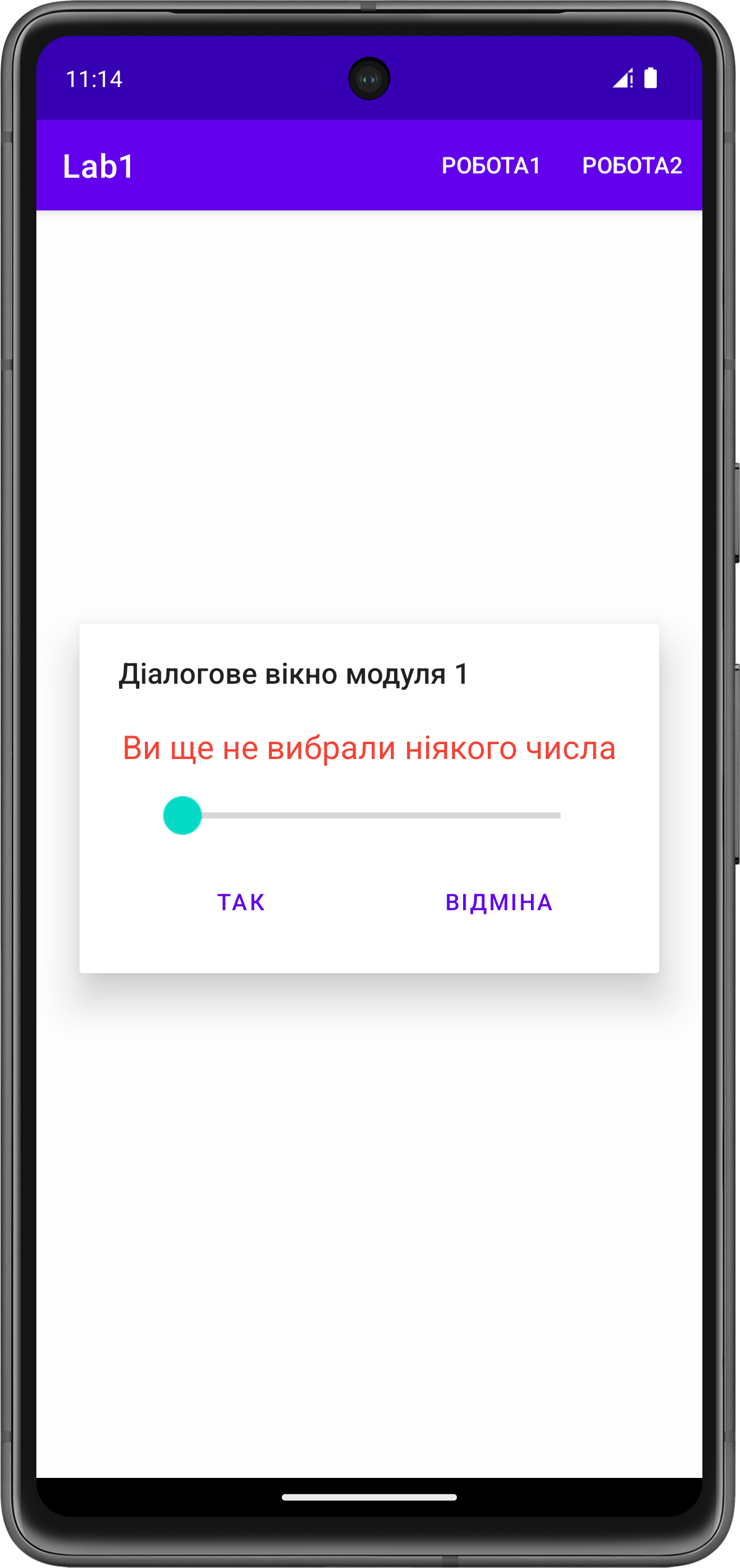
package com.oop.lab1.module3  
  
import android.content.Context  
import androidx.fragment.app.FragmentManager  
import com.oop.lab1.Module3Handler  
interface Module3Interface {  
 fun run(  
 context: Context,  
 manager: FragmentManager,  
 handler: Module3Handler  
 )  
}

**Ілюстрації виконання програми**

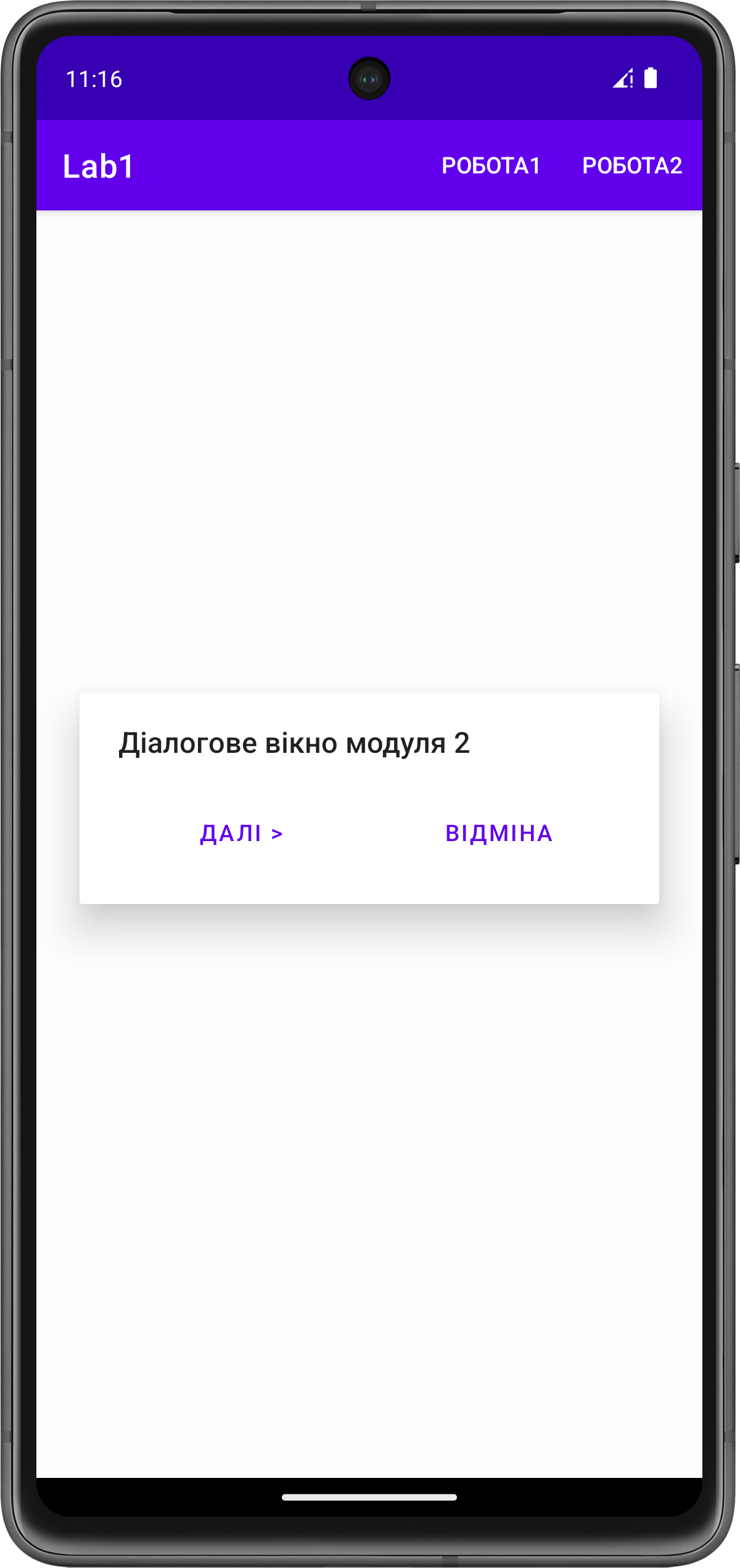
**Вікно головного модуля *Lab1:***



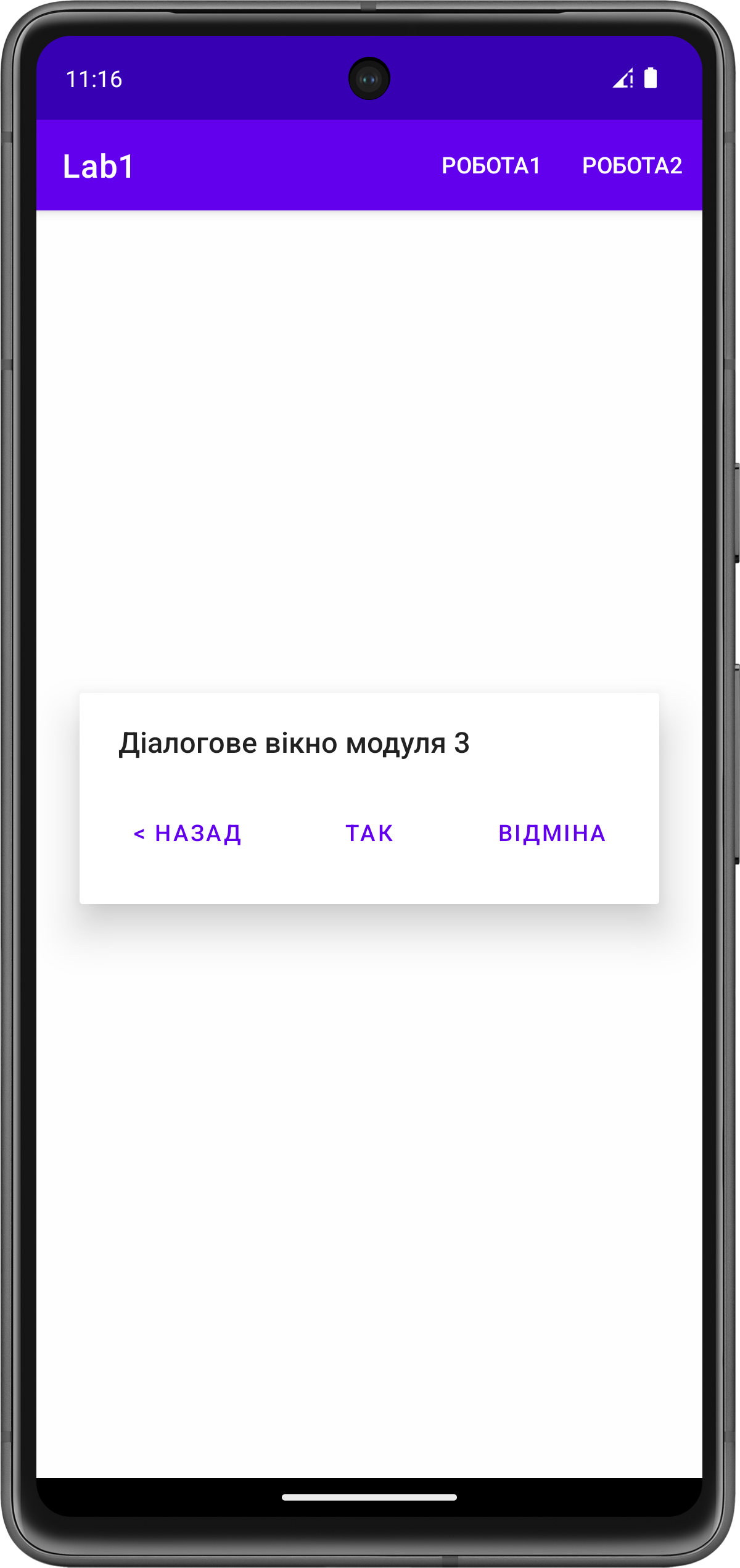
**Виконання модуля *Module1:***



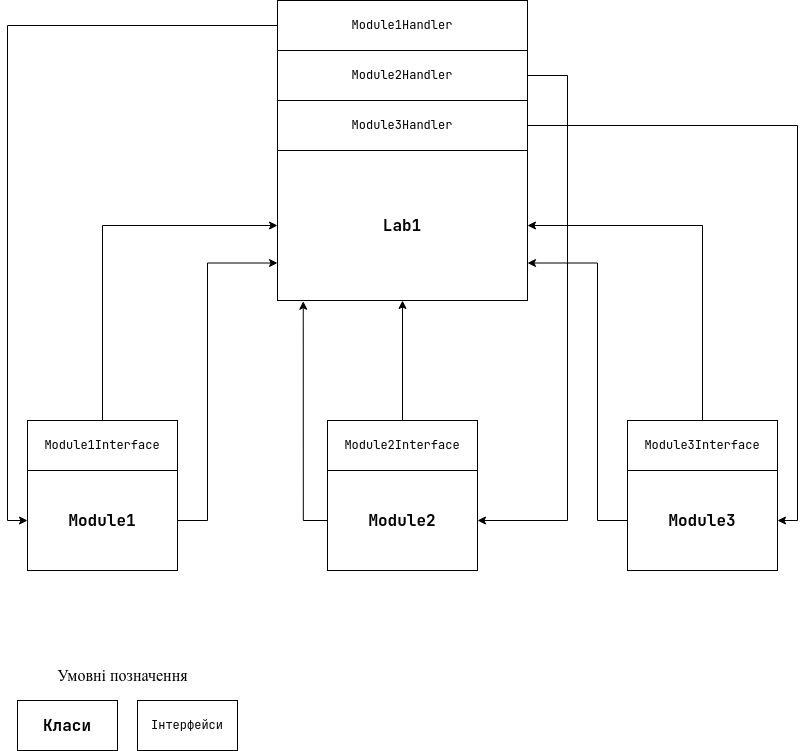
**Виконання модуля *Module2:***



**Виконання модуля *Module3:***



**Діаграма import-ієрархії програми**

****

**Висновки**

Під час виконання цієї лабораторної роботи я отримав базові навички створення віконних додатків для платформи ***Android***. У якості інтегрованого середовища розробки я обрав ***Android Studio***, а мови програмування – ***Kotlin***. Завдяки здобутим знанням мені вдалося створити простий проєкт, що складається з головного модуля та трьох незалежних діалогових вікон.