

**Міністерство освіти і науки України  
Національний технічний університет України "Київський політехнічний  
інститут імені Ігоря Сікорського"  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

**ЗВІТ**

про виконання лабораторної роботи №3.1

з дисципліни

**«Інтелектуальні вбудовані системи»**

на тему:

**«РЕАЛІЗАЦІЯ ЗАДАЧІ РОЗКЛАДАННЯ ЧИСЛА НА ПРОСТІ  
МНОЖНИКИ (ФАКТОРИЗАЦІЯ ЧИСЛА)»**

Перевірів:  
асистент кафедри ОТ  
Регіда П. Г

ВИКОНАВ:  
студент 3 курсу  
групи ІП-83, ФІОТ  
Валігура М.І.  
Залікова книжка №8303  
Варіант – 3

Київ 2021

## Завдання на лабораторну роботу

Розробити програму для факторизації заданого числа методом Ферма.  
Реалізувати користувацький інтерфейс з можливістю вводу даних.

Програмний код:

fermat\_factorization.dart

```
import 'dart:math';

String fermatFactorization(dynamic inputValue) {
  if (inputValue.isEmpty) return 'Input value must be an integer';
  final inputNumber = int.parse(inputValue);
  if ((inputNumber & 1) == 0) {
    return '${inputNumber / 2} * 2';
  }
  var a = sqrt(inputNumber).ceil();
  if (a * a == inputNumber) return '$a * $a';
  var b;
  while (true) {
    int b1 = a * a - inputNumber;
    b = sqrt(b1).ceil();
    if (b * b == b1)
      break;
    else
      a += 1;
  }
  return '${a - b} * ${a + b}';
}
```

fermat\_factorization\_screen.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import '../helpers/fermat_factorization.dart';

class FermatFactorization extends StatefulWidget {
  @override
  _FermatFactorizationState createState() => _FermatFactorizationState();
}

class _FermatFactorizationState extends State<FermatFactorization> {
  final _numberController = TextEditingController();
  String resultValue = '';
  bool _offstage = true;

  @override
  void dispose() {
    _numberController.dispose();
    super.dispose();
  }
}
```

```

@override
Widget build(BuildContext context) {
  return Container(
    child: Padding(
      padding: const EdgeInsets.all(80.0),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          TextField(
            controller: _numberController,
            decoration: new InputDecoration(labelText: "Enter your number"),
            autocorrect: true,
            keyboardType: TextInputType.number,
            inputFormatters: [FilteringTextInputFormatter.digitsOnly],
          ),
          Offstage(
            offstage: _offstage,
            child: Padding(
              padding: const EdgeInsets.all(25.0),
              child: Text(resultValue),
            ),
          ),
          _offstage ? SizedBox(height: 8.0) : SizedBox(height: 0.0),
          ElevatedButton(
            child: const Text('Calculate'),
            onPressed: () {
              setState(() {
                _offstage = false;
                resultValue = fermatFactorization(_numberController.text);
              });
            },
          ),
        ],
      ),
    ),
  );
}

```

main\_screen.dart

```

import 'package:flutter/material.dart';
import 'package:lab3_mobile/screens/fermat_factorization_screen.dart';
import 'package:lab3_mobile/widgets/bottom_navigation.dart';

List<Widget> _options = <Widget>[
  FermatFactorization(),
  FermatFactorization(),
  FermatFactorization(),
];

```

```

class MainScreen extends StatefulWidget {
  @override
  _MainScreenState createState() => _MainScreenState();
}

class _MainScreenState extends State<MainScreen> {
  int _currentIndex = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Center(
          child: Text(allTabItems[_currentIndex].title),
        ),
      ),
      body: _options.elementAt(_currentIndex),
      bottomNavigationBar: BottomNavigationBar(
        currentIndex: _currentIndex,
        onSelectTab: (int index) {
          setState(() {
            _currentIndex = index;
          });
        },
      ),
    );
  }
}

```

bottom\_navigation.dart

```

import 'package:flutter/material.dart';

class TabItem {
  TabItem({this.label, this.title, this.icon, this.backgroundColor});
  final String label;
  final String title;
  final Icon icon;
  final Color backgroundColor;
}

List<TabItem> allTabItems = <TabItem>[
  TabItem(
    icon: Icon(Icons.miscellaneous_services),
    label: 'factorization',
    title: 'Fermat`s factorization example',
    backgroundColor: Colors.redAccent[400]),
  TabItem(
    icon: Icon(Icons.more_time),
    label: 'lab32',
    title: 'lab32',
  ),
]

```

```

        backgroundColor: Colors.tealAccent[400]),
    TabItem(
      icon: Icon(Icons.more_time),
      label: 'lab33',
      title: 'lab33',
      backgroundColor: Colors.deepPurpleAccent[400]),
  ];

class BottomNavigation extends StatelessWidget {
  BottomNavigation({@required this.currentIndex, @required this.onSelectTab});

  final int currentIndex;
  final ValueChanged<int> onSelectTab;

  @override
  Widget build(BuildContext context) {
    return BottomNavigationBar(
      type: BottomNavigationBarType.shifting,
      items: allTabItems
        .map((TabItem tabItem) => BottomNavigationBarItem(
          icon: tabItem.icon,
          backgroundColor: tabItem.backgroundColor,
          label: tabItem.label))
        .toList(),
      currentIndex: currentIndex,
      selectedItemColor: Colors.white,
      unselectedItemColor: Colors.grey,
      onTap: onSelectTab,
      iconSize: 30,
    );
  }
}

```

main.dart

```

import 'package:flutter/material.dart';
import 'screens/main_screen.dart';

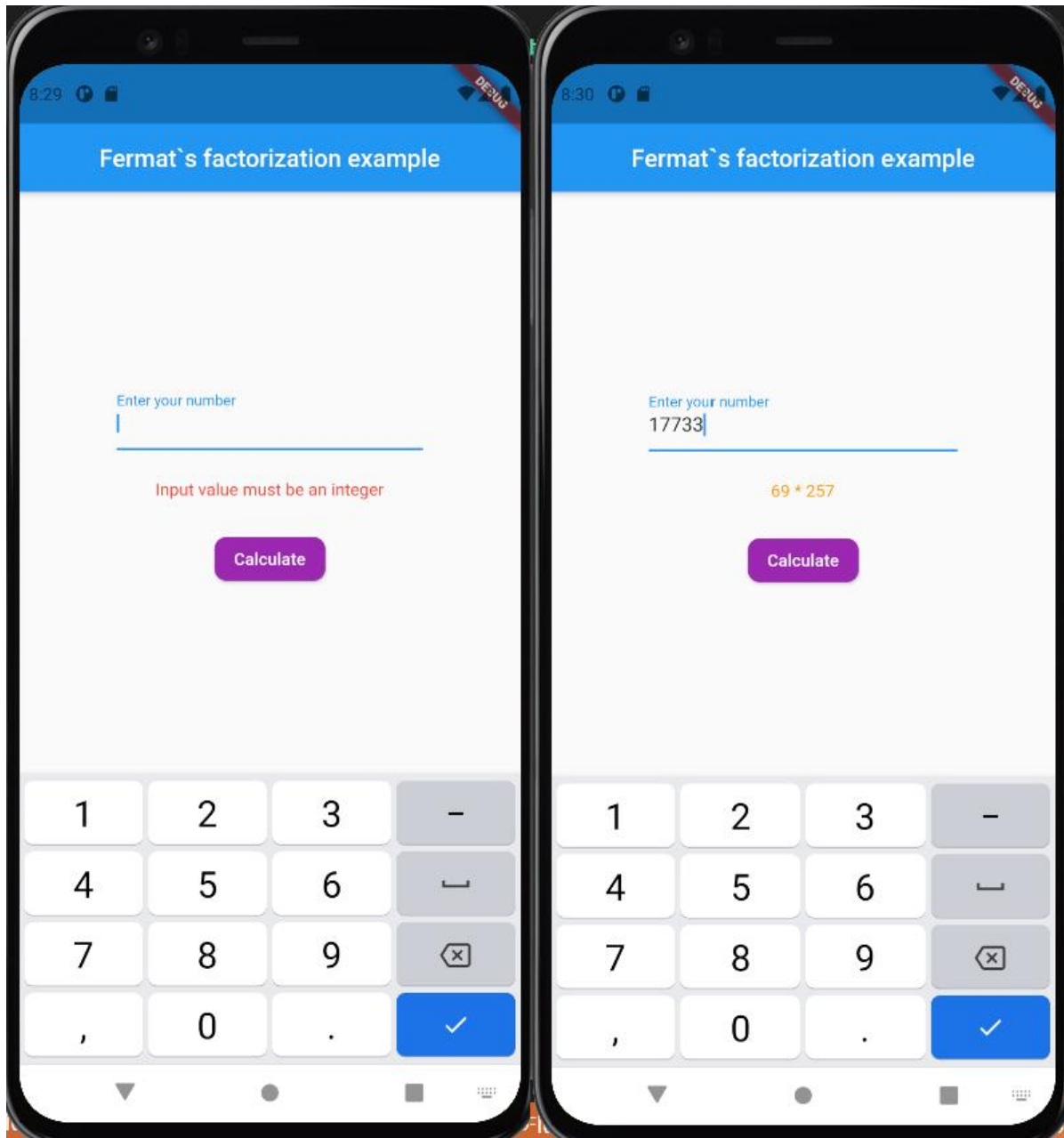
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        primarySwatch: Colors.blue,
        visualDensity: VisualDensity.adaptivePlatformDensity,

```

```
    ),  
    home: MainScreen(),  
  );  
}  
}
```

Результати роботи програми:



Висновки:

Отже, в ході лабораторної роботи, ми ознайомилися з принципами розкладання числа на прості множники з використанням алгоритму факторизації Ферма. Результати наведено в скріншотах. Мету було досягнуто.