Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

ЗВІТ

до лабораторної роботи №3.1

з дисципліни «Інтелектуальні вбудовані системи»

на тему «Реалізація задачі розкладання числа на прості множники (факторизація числа)»

Виконав:
студент групи ІП-83
Черевач А.М.

Перевірив:
асистент Регіда П.Г.

Київ - 2021

Основні теоретичні відомості

Факторизації лежить в основі стійкості деяких криптоалгоритмів, еліптичних кривих, алгебраїчній теорії чисел та кванових обчислень, саме тому дана задача дуже гостро досліджується, й шукаються шляхи її оптимізації.

На вхід задачі подається число n Є N, яке необхідно факторизувати. Перед виконанням алгоритму слід переконатись в тому, що число не просте. Далі алгоритм шукає перший простий дільник, після чого можна запустити алгоритм заново, для повторної факторизації.

В залежності від складності алгоритми факторизації можна розбити на дві групи:

- Експоненціальні алгоритми (складність залежить експоненційно від довжини
  вхідного параметру);
- Субекспоненціальні алгоритми.

Існування алгоритму з поліноміальною складністю – одна з найважливіших проблем в сучасній теорії чисел. Проте, факторизація з даною складністю можлива на квантовому комп'ютері за допомогою алгоритма Шора.

Метод факторизації Ферма.

Ідея алгоритму заключається в пошуку таких чисел A і B, щоб факторизоване число n мало вигляд: n = A2 – B2. Даний метод гарний тим, що реалізується без використання операцій ділення, а лише з операціями додавання й віднімання.

Приклад алгоритму:

Початкова установка: x = [$\sqrt{n}$] – найменше число, при якому різниця x2-n невід'ємна. Для кожного значення k Є N, починаючи з k = 1, обчислюємо ($\lceil\sqrt{n}\rceil + k$)2 − n і перевіряємо чи не є це число точним квадратом.

Якщо не є, то k++ і переходимо на наступну ітерацію.

Якщо є точним квадратом, тобто $x2 - n = (\lceil \sqrt{n} \rceil + k)2 - n = y2$, то ми отримуємо розкладання: $n = x2 - y2 = (x + y)(x - y) = A * B$, в яких

$x = (\lceil \sqrt{n} \rceil + k)$

Якщо воно є тривіальним і єдиним, то n - просте

Завдання

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

Лістинг програми

```
import React, { useState } from 'react';

import { StyleSheet, Text, View, SafeAreaView, TextInput, Button } from
'react-native';



export default function App() {

 const ferma = (n) => {

   if (n < 1 || !/^[0-9]*$/.test(n))

     return "N value is invalid";



   if (n % 2 == 0)

     return `A: ${n / 2}, B: 2`;



   let x = ~~(Math.sqrt(n));

   let y = 0;



   while (true) {
```

```
    const squaredY = x ** 2 - n;
    y = ~~(Math.sqrt(squaredY));

    if (y ** 2 === squaredY)
      break;
    else
      x += 1
  }

  return `A: ${x - y}, B: ${x + y}`;
}

const [n, onChangeNumber] = useState(null);
const [result, setResult] = useState(null);
const pressHandler = () => setResult(ferma(n));

return (
  <SafeAreaView>
    <TextInput
      style={styles.input}
      onChangeText={onChangeNumber}
      value={n}
      placeholder="Write n value"
      keyboardType="numeric"
    />
    <View style={styles.btn}>
```

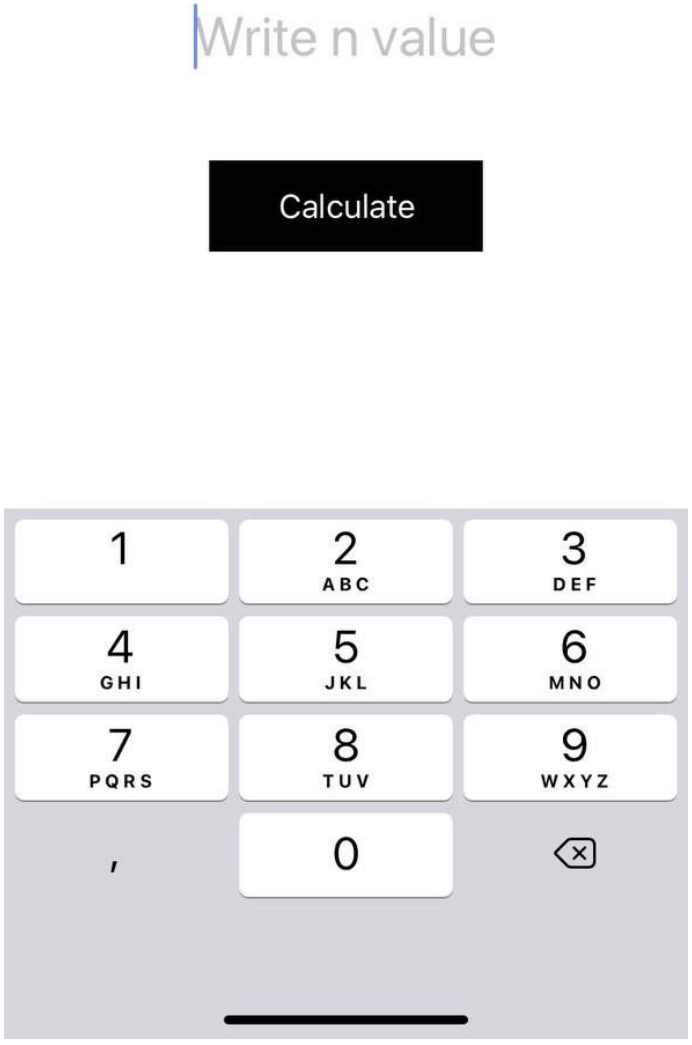
```
      <Button
        title="Calculate"
        color="#fff"
        onPress={pressHandler}
      />
    </View>
    <Text style={styles.result}>
      {result}</Text>
  </SafeAreaView>
 );
};



const styles = StyleSheet.create({
 container: {
   flex: 1,
   backgroundColor: '#fff',
   alignItems: 'center',
   justifyContent: 'center',
 },
 input: {
   alignSelf: 'center',
   top: 200,
   fontSize: 30
 },
 btn: {
```

```
    justifyContent: 'center',
    alignItems: 'center',
    alignSelf: 'center',
    top: 250,
    height: 50,
    width: 150,
    backgroundColor: 'black',
  },
  result: {
    alignSelf: 'center',
    top: 300,
    fontSize: 25
  }
});
```

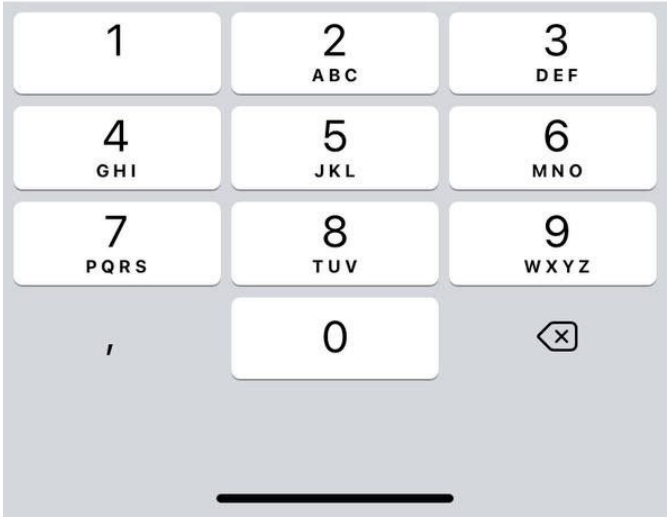
Результат роботи програми

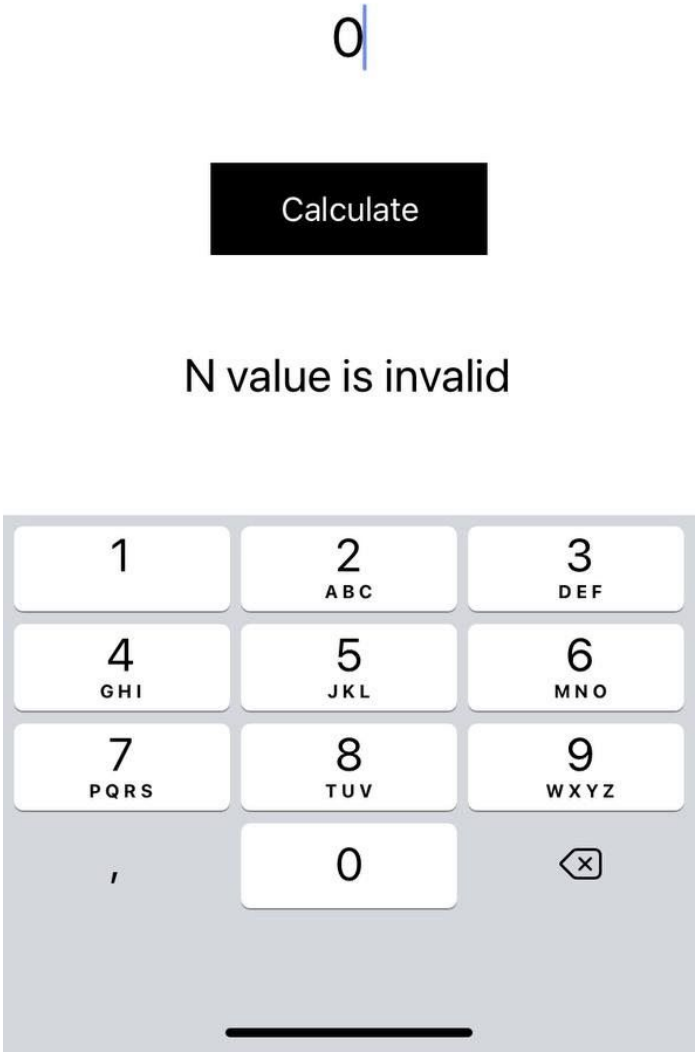


33345

Calculate

A: 171, B: 195




0
Calculate
N value is invalid
1
2 ABC
3 DEF
4 GHI
5 JKL
6 MNO
7 PQRS
8 TUV
9 WXYZ
,
0

Висновки

Під час виконання лабораторної роботи я дослідив метод факторизації числа Ферма. Було реалізовано програму для розкладання числа на прості множники у вигляді мобільного додатку за допомогою фреймворку React Native та Expo. Програма розкладає введене число на два простих множника та виводить їх на екран.