Object Oriented Programming (CSC 2510)

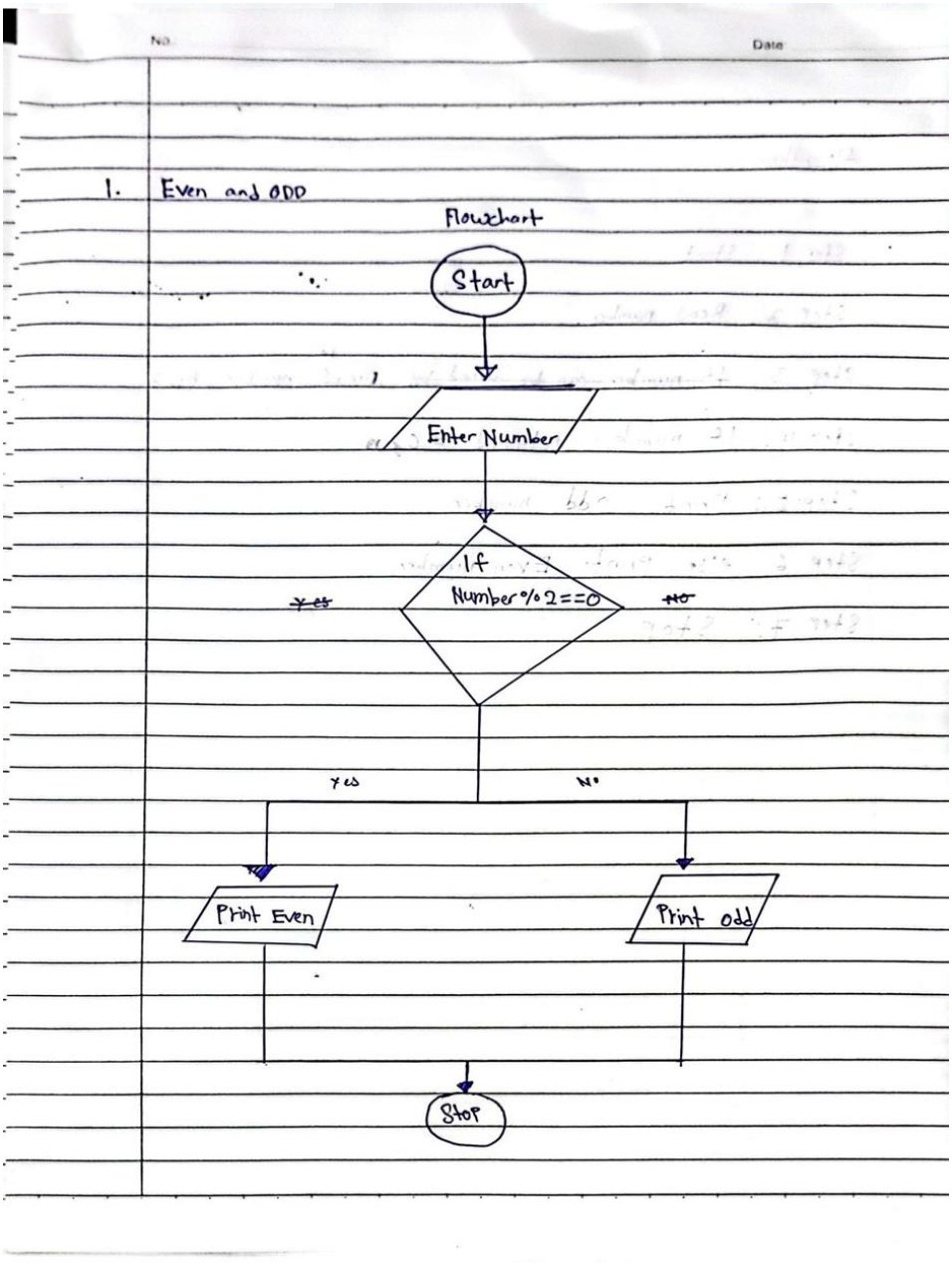
ASSIGNMENT 1

Full Name	:	Donovan Jude
Matric No.	:	221023210
Semester	:	June 2023
Submission Date	:	16 August 2023

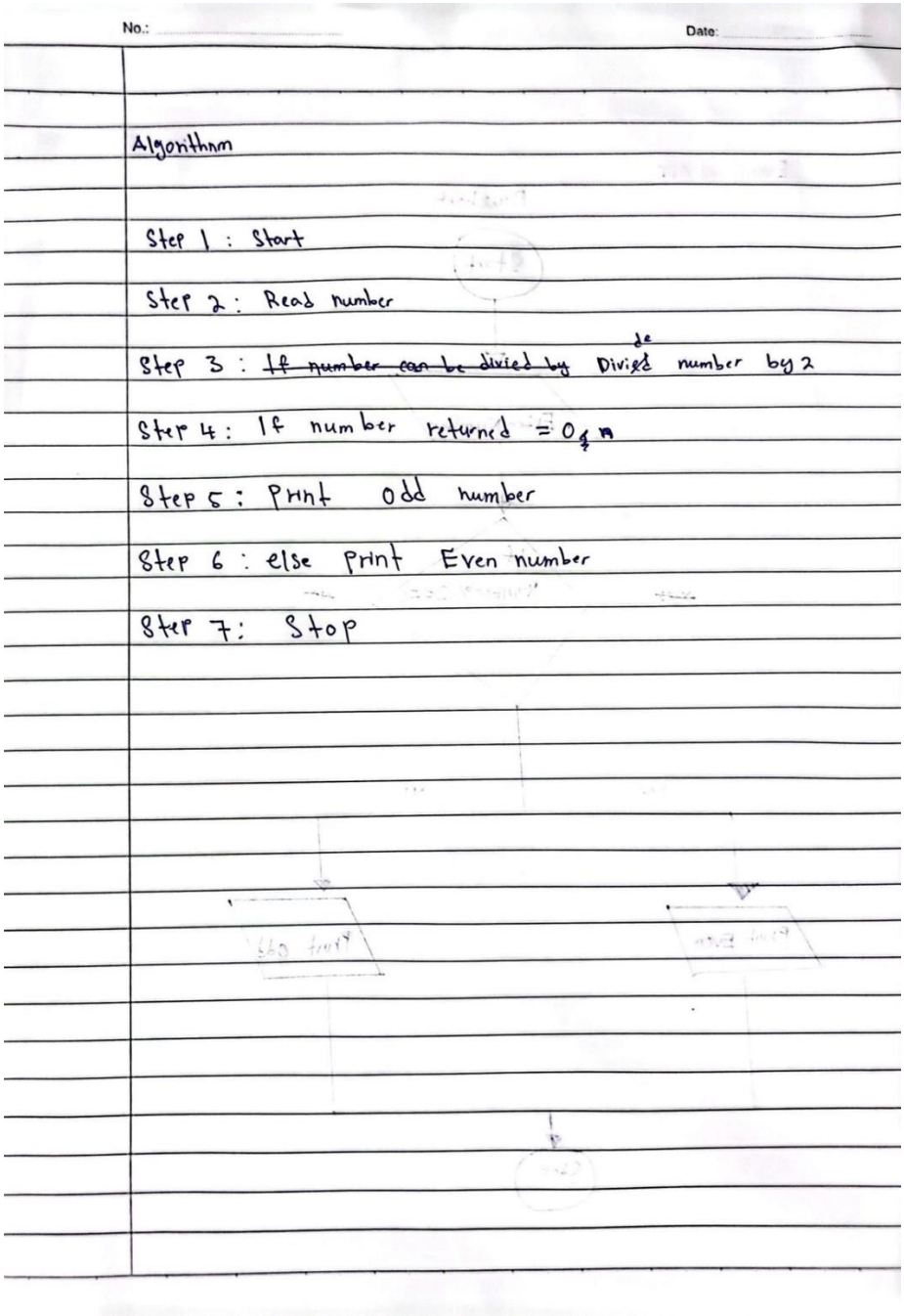
Marks Allocation		
Flowchart& Algorithm	:	
Code	:	
Screenshots	:	
Format	:	
TOTAL MARKS	:	/30

DEVELOP ALGORITHM AND FLOW CHART FOR ALL FUNCTIONS

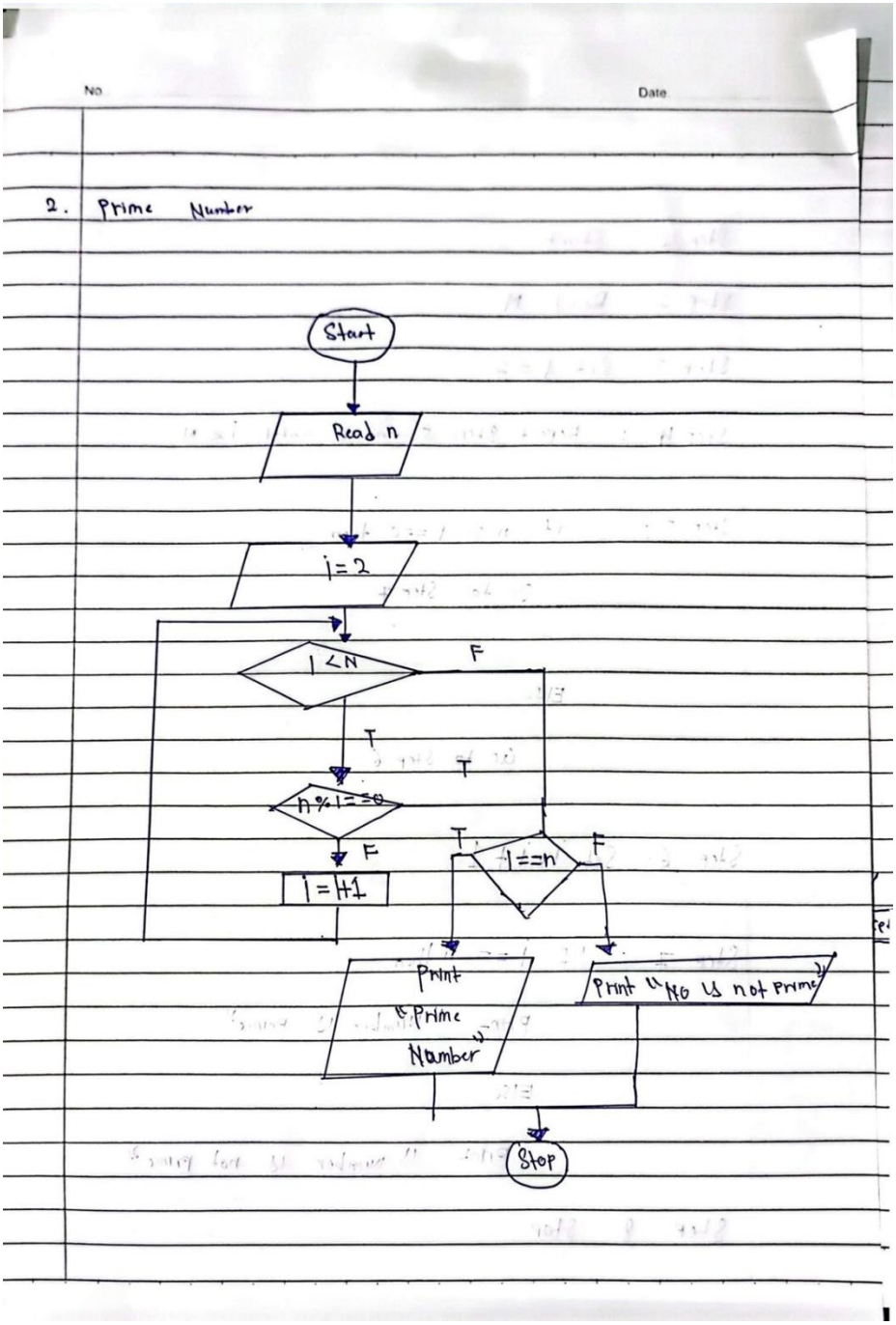
1. EVEN&ODD Flowchart



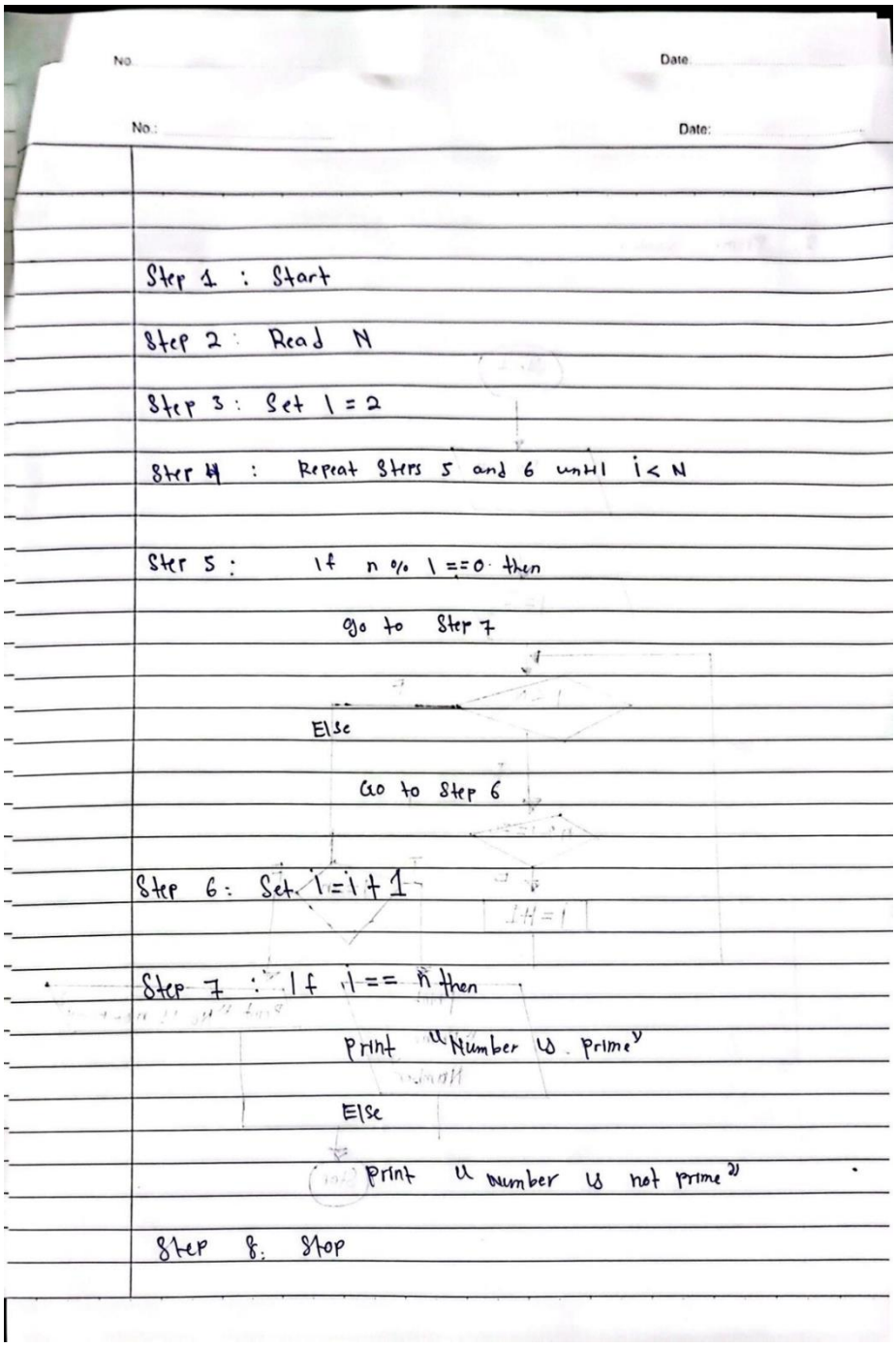
1.1 EVEN&ODD Algorithm



1.2 PRIME Flowchart



1.2 PRIME Algorithm



No. Date:

No.: Date:

Step 1 : Start

Step 2 : Read N

Step 3 : Set i = 2

Step 4 : Repeat Steps 5 and 6 until i < N

Step 5 : If n % i == 0 then

go to Step 7

Else

Go to Step 6

Step 6 : Set i = i + 1

Step 7 : If i == n then

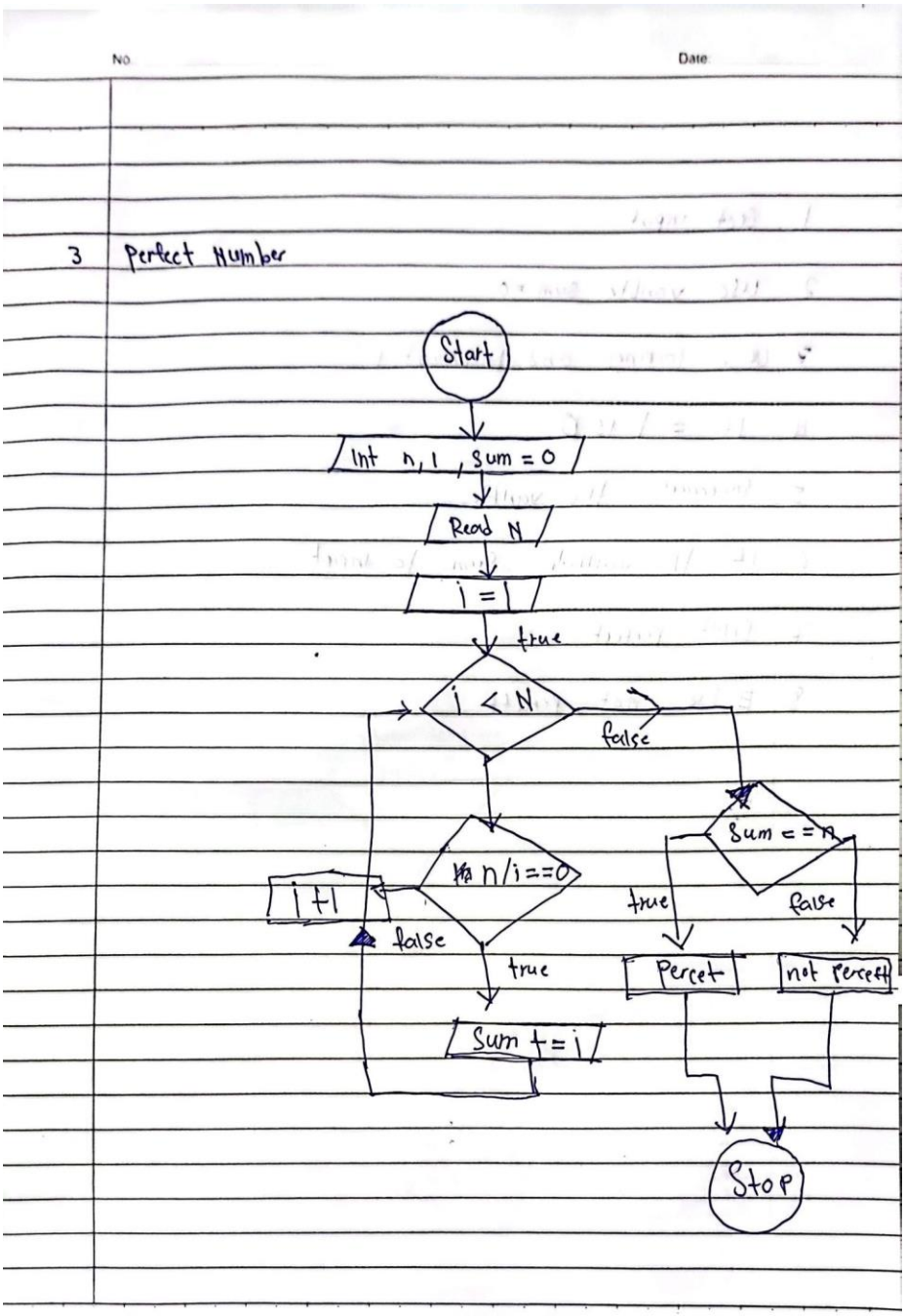
Print "Number is prime"

Else

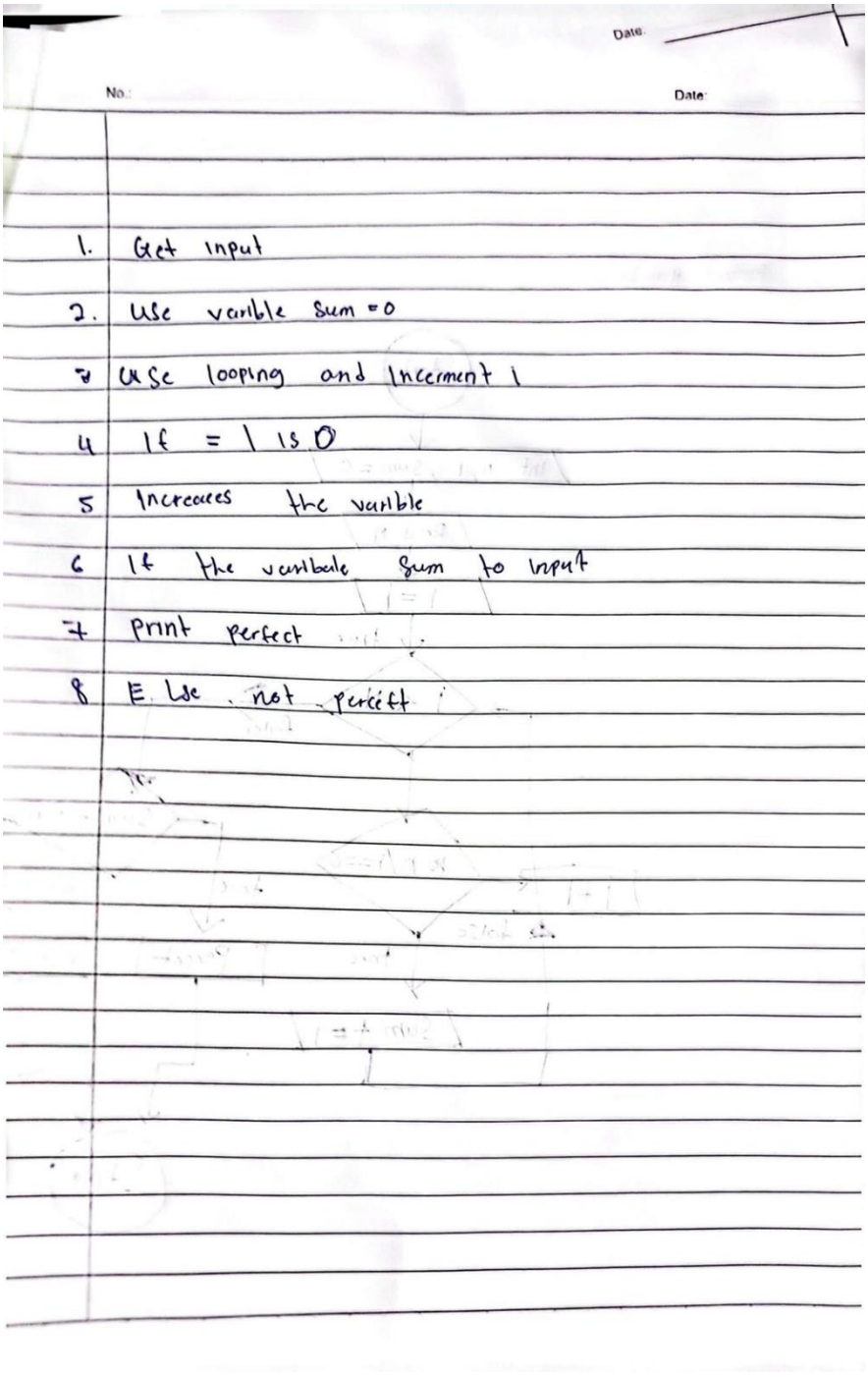
Print "number is not prime"

Step 8 : Stop

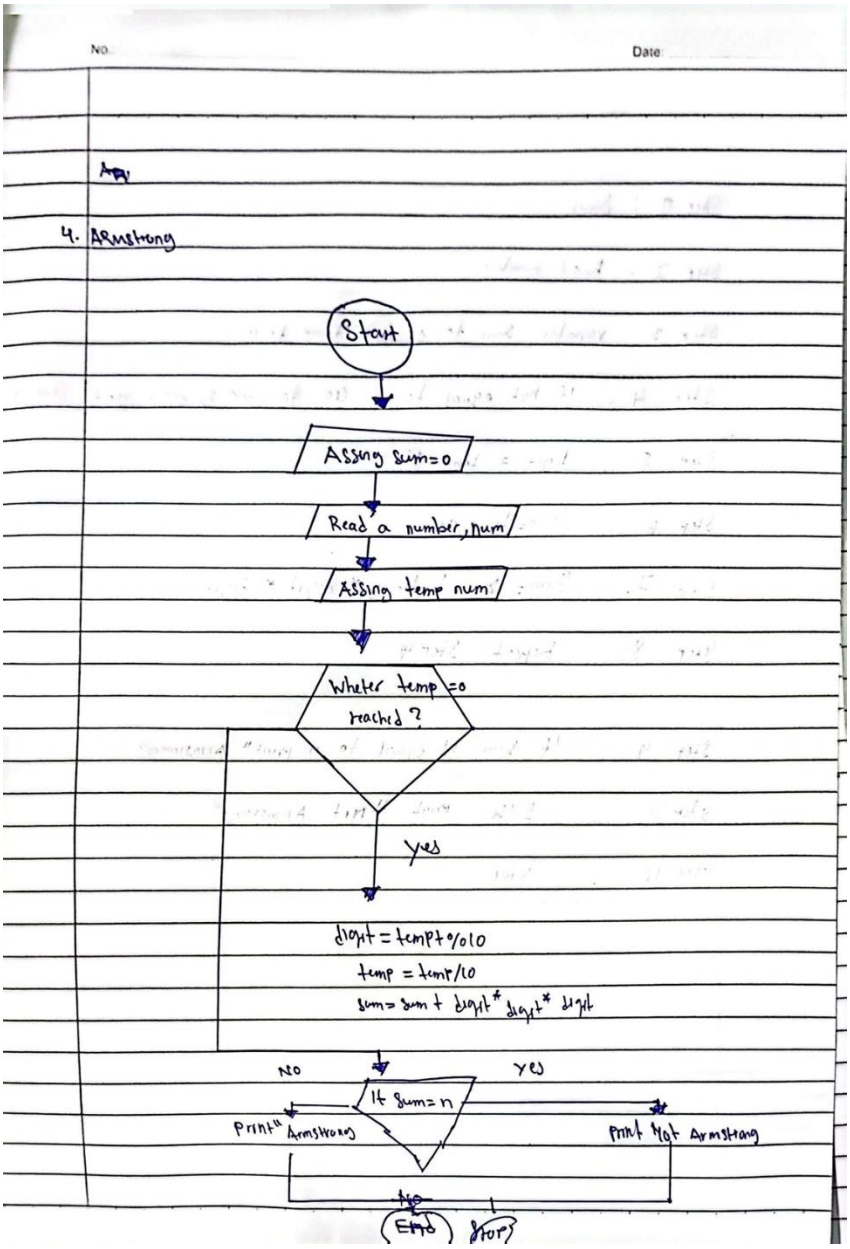
1.3 PERFECT Flowchart



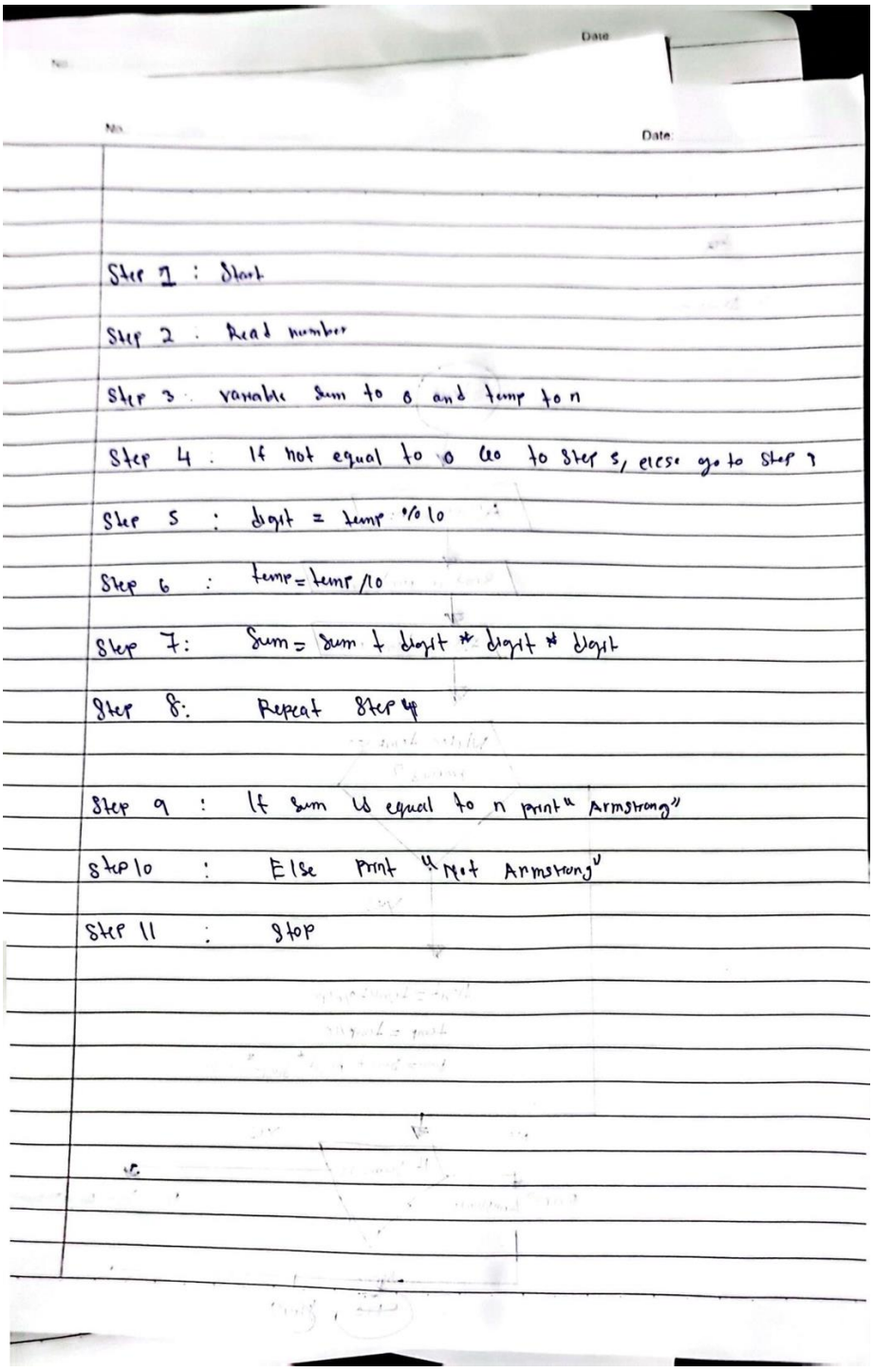
1.3 PERFECT Algorithm



1.4 ARMSTRONG Flowchart



1.4 ARMSTRONG Algorithm



Step 1 : Start

Step 2 : Read number

Step 3 : variable sum to 0 and temp to n

Step 4 : If not equal to 0 go to step 5, else go to step 7

Step 5 : digit = temp % 10

Step 6 : temp = temp / 10

Step 7: Sum = sum + digit * digit * digit

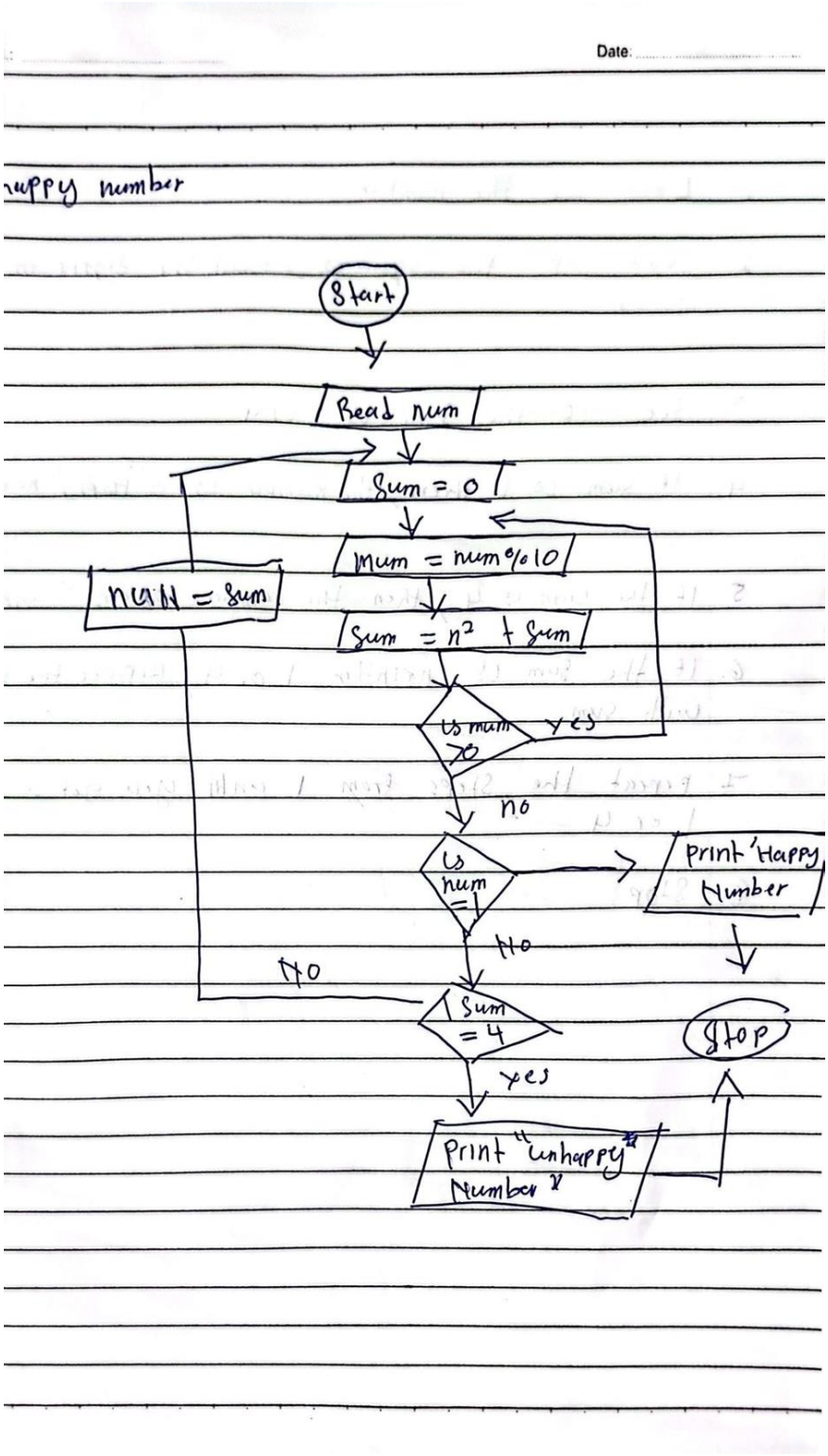
Step 8: Repeat Step 4

Step 9 : If sum is equal to n print " Armstrong"

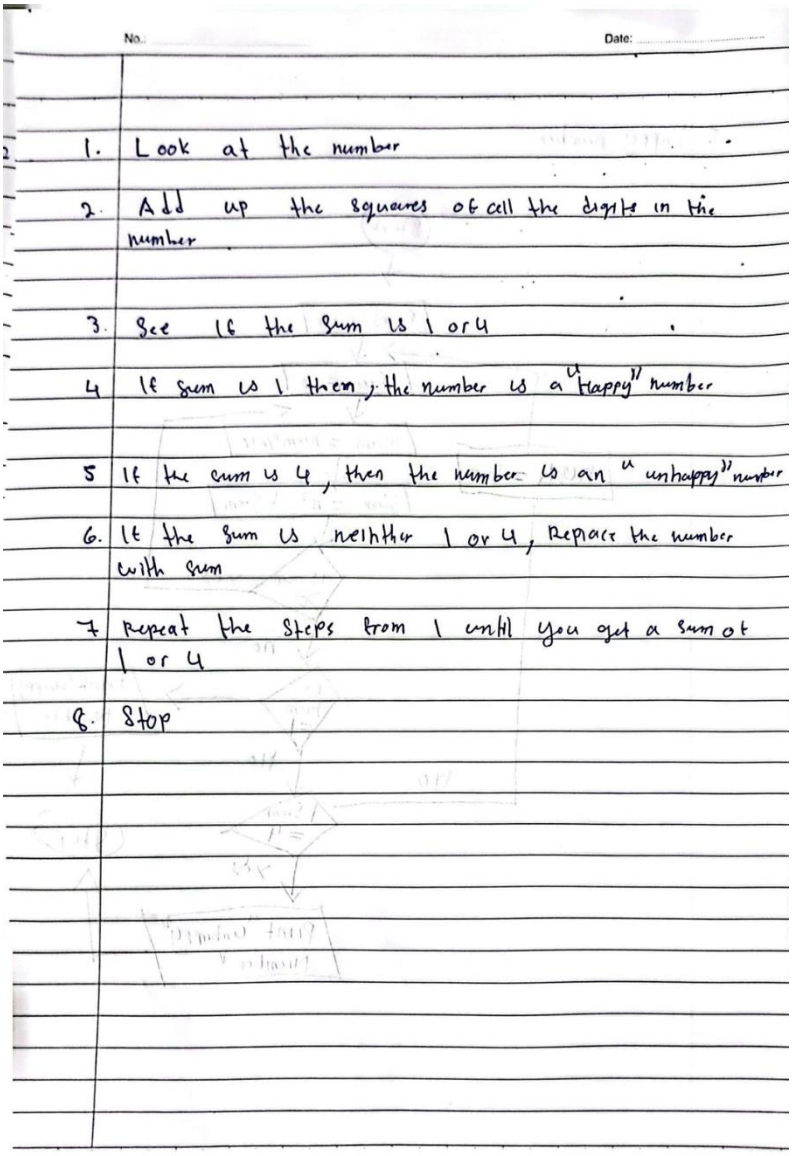
Step 10 : Else print "Not Armstrong"

Step 11 : Stop

1.5 HAPPY FLOWCHART



1.5 HAPPY **Algorithm**



No. Date:

1. Look at the number
2. Add up the squares of all the digits in the number
3. See if the sum is 1 or 4
4. If sum is 1 then, the number is a "Happy" number
5. If the sum is 4, then the number is an "unhappy" number
6. If the sum is neither 1 or 4, Replace the number with sum
7. Repeat the steps from 1 until you get a sum of 1 or 4
8. Stop

2.1 Implement functions as separate METHODS and name file as Numbers.java.

Code package numbers.java0; import java.util.Scanner;

```
/**
 *
 * @author Don
 */ public class DJ {

   public static void main(String[] args) {
      Scanner kb = new Scanner(System.in);        System.out.println("Enter a Number =");
      int input = kb.nextInt();          DJ obj = new DJ();          obj.even(input);          obj.odd(input);
obj.prime(input);        obj.perfect(input);        obj.armstrong(input);        obj.happy(input);
   }

   void even(int input) {

      if (input % 2 == 0) {
         System.out.println(input + " is even");
      } else {
         System.out.println(input + " is not even");
      }

   }

   void odd(int input) {        if (input % 2 == 1) {
         System.out.println(input + " is odd");
      } else {
         System.out.println(input + " is not odd");
      }
   }

   void prime(int input) {

      int i, count = 0;        for (i = 1; i <= input; i++) {          if (input % i == 0) {              count++;
         }
      }
      if (count == 2) {
         System.out.println(input + " is prime");
      } else {
         System.out.println(input + " is not prime");
      }
   }
```

```
    void perfect(int input) {

        int perf, sum = 0;

        for (int i = 1; i < input; i++) {            if (input % i == 0) {                sum = sum + i;
            }
        }
        if (sum == input) {
            System.out.println(input + " is perfect ");
        } else {
            System.out.println(input + " not perfect");
        }
    }

    void armstrong(int input) {

        int sum = 0;        int backup = input;

        while (input > 0) {

            sum = sum + (input % 10) * (input % 10) * (input % 10);
            input = input / 10;

        }
        if (sum == backup) {
            System.out.println(backup + " is armstrong");
        } else {
            System.out.println(backup + " is not armstrong");
        }

    }

    void happy(int input) {
        int sum = 0;        int rem, num = 1;        int input1 = input;        while (num != 0) {            rem =
input % 10;

            sum += (rem * rem) ;

                input = input / 10;                num = input;


        }
        if (sum == 1) {
```

```
            System.out.println(input1 + "is a Happy number");
        } else {
            System.out.println(input1 + "is a not a Happy number");
        }

}
}
```

Output:

```
run:
Enter a Number =
1
1 is not even
1 is odd
1 is not prime
1 not perfect
1 is armstrong
1is a Happy number
BUILD SUCCESSFUL (total time: 4 seconds)
```

3.1 Get multiple inputs from the user to test. Hint: User array to store inputs and output

Code:

```
public class Numbers {
  public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
while(true){

    System.out.println("Choose From Below");
    System.out.println("1. User Input");
    System.out.println("2. Array input");
    int selection=0;
    try{
     selection = kb.nextInt();
    }catch (InputMismatchException e) {
        System.out.println("Error: Size must be an integer. Please try again.");
        kb.next();
        continue;
      }

  if (selection == 1) {

      System.out.println("Size of the array?");
      int size = 0;
      try {
        size = kb.nextInt();
      } catch (InputMismatchException e) {
        System.out.println("Error:Please try again");
        kb.next();
        continue;
      }

      int[] numbers = new int[size];
      for (int i = 0; i < size; i++) {
        System.out.printf("Enter integer %d = ", i + 1);
```

```
if (selection == 1) {

    System.out.println("Size of the array?");
    int size = 0;
    try {
      size = kb.nextInt();
    } catch (InputMismatchException e) {
      System.out.println("Error:Please try again");
      kb.next();
      continue;
    }

    int[] numbers = new int[size];
    for (int i = 0; i < size; i++) {
      System.out.printf("Enter integer %d = ", i + 1);
      int number = 0;
      try {
        number = kb.nextInt();
      } catch (InputMismatchException e) {
        System.out.println("Error:Please try again");
        kb.next();
        i--;

      }
      numbers[i] = number;
    }

    // Check the numbers in the array
    checkNumbers(numbers);
  } else if (selection == 2) {
    // Range input
    // Get the range from the user
    System.out.println("Enter your range?");
    int range = 0;
    try {
      range = kb.nextInt();
```

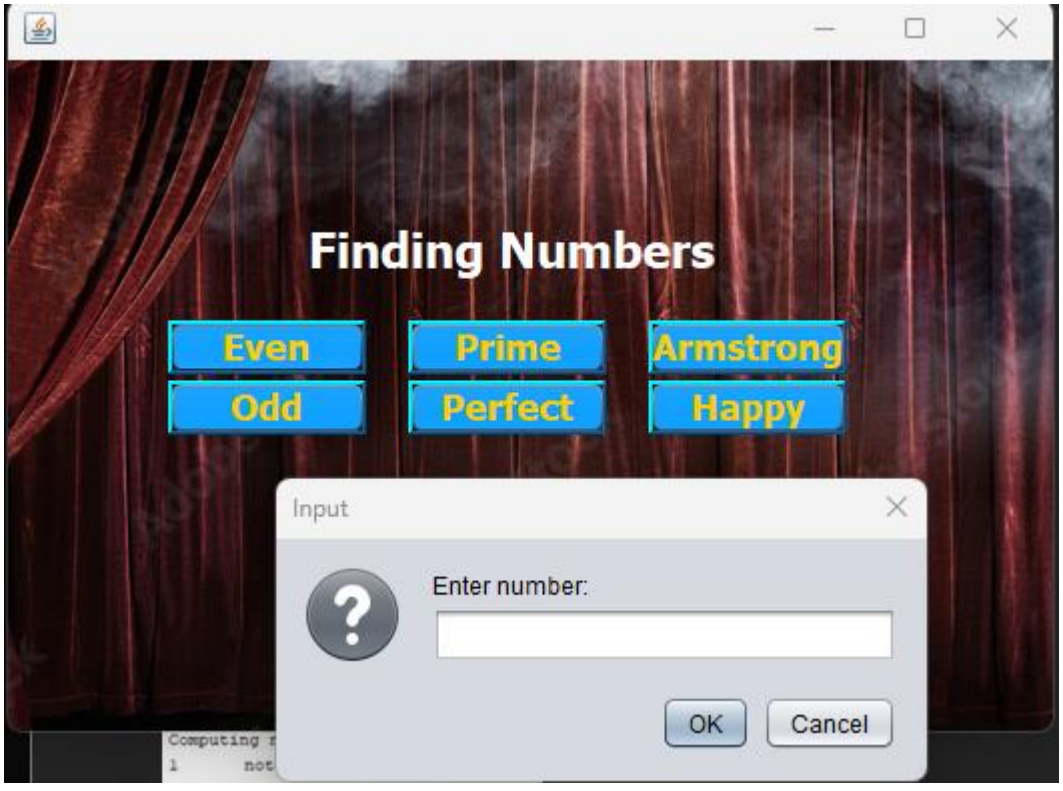
Output:

```
run:
Choose From Below
1. User Input
2. Array input
2
Enter your range?
4
Computing number: 1
1       not even
1       odd
1       Not prime
1       Armstrong
1       Not perfect
1       Happy
Computing number: 2
2       Even
2       not odd
2       Prime
2       Not Armstrong
2       Not perfect
2       Not happy
Computing number: 3
3       not even
3       odd
3       Prime
3       Not Armstrong
3       Not perfect
3       Not happy
Computing number: 4
4       Even
4       not odd
4       Not prime
4       Not Armstrong
4       Not perfect
4       Not happy
Choose From Below
1. User Input
2. Array input
```

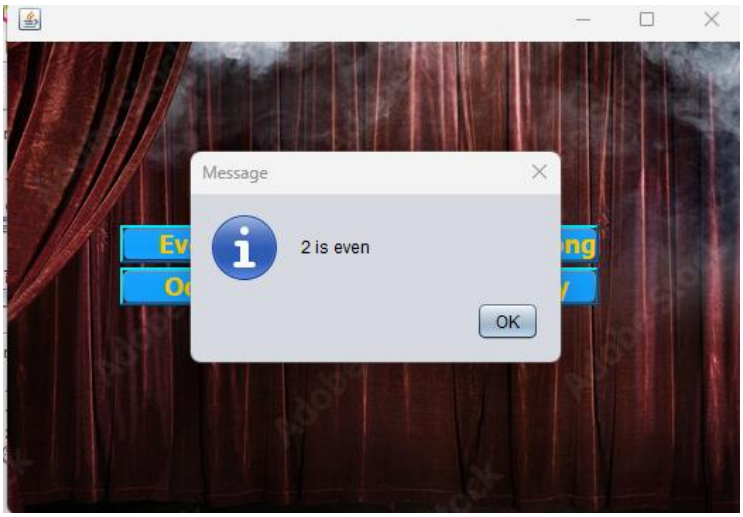
4.0 GUI

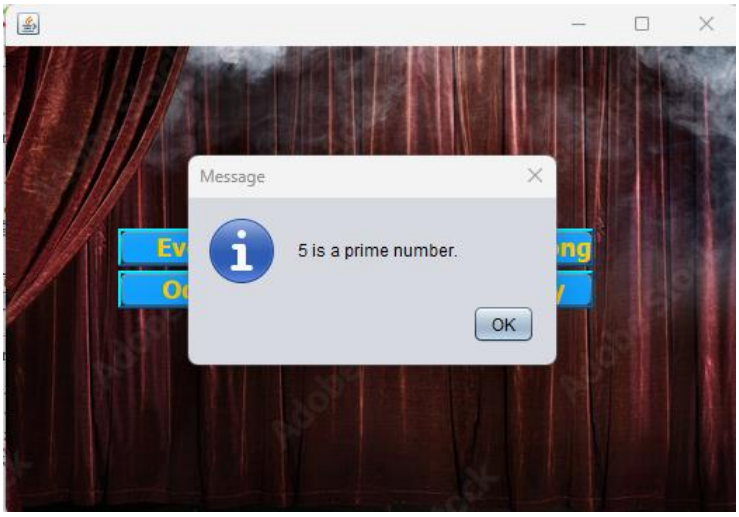
Input :

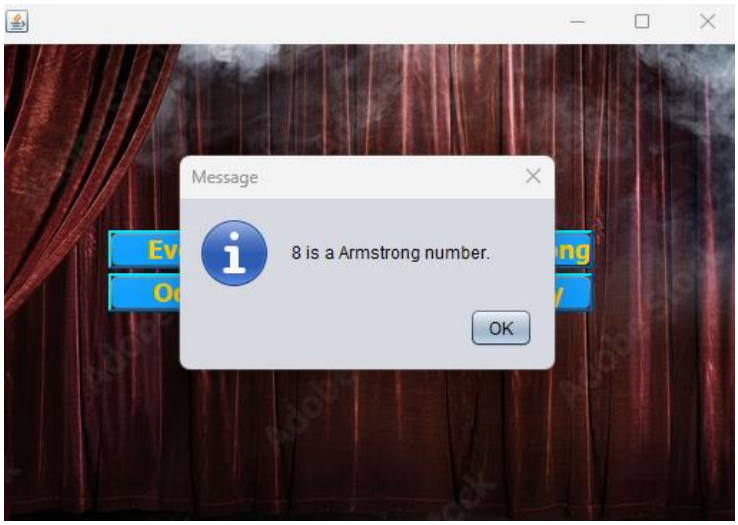

Finding Numbers
Even
Prime
Armstrong
Odd
Perfect
Happy

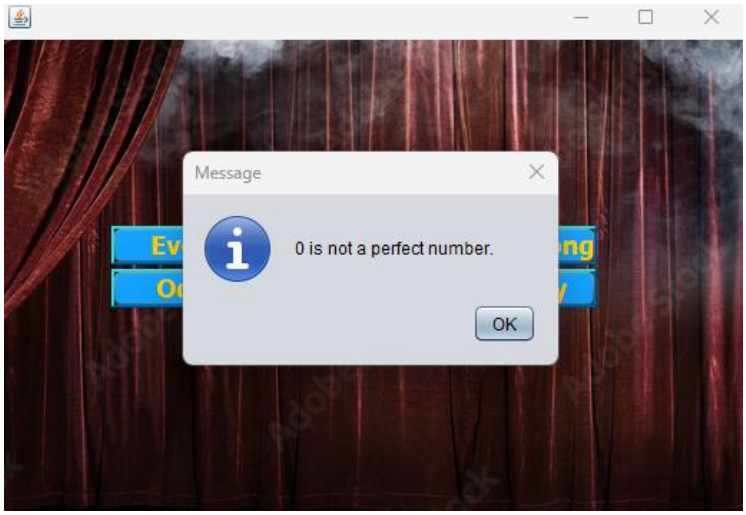

Finding Numbers
Even
Prime
Armstrong
Odd
Perfect
Happy
Input
Enter number:
OK
Cancel

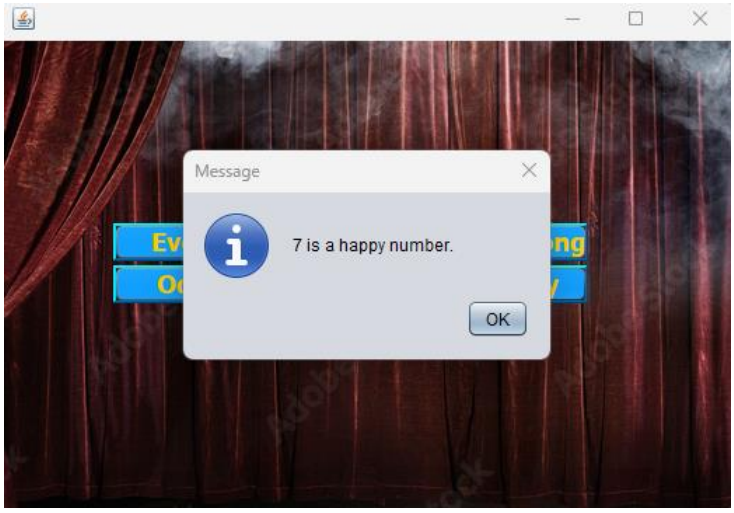

Finding Numbers
Even
Prime
Armstrong
Odd
Perfect
Happy
Message
9 is odd
OK


Message
2 is even
OK


Message
5 is a prime number.
OK


Message
8 is a Armstrong number.
OK


Message
0 is not a perfect number.
OK


Message
7 is a happy number.
OK

Source code

```
import java.util.HashSet;
import javax.swing.JOptionPane;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Don
 */
public class Gui extends javax.swing.JFrame {

    /**
     * Creates new form Gui
     */
    public Gui() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        Btneven = new javax.swing.JButton();
        BtnOdd = new javax.swing.JButton();
        btnPrime = new javax.swing.JButton();
        jButton1 = new javax.swing.JButton();
        btnArmstrong = new javax.swing.JButton();
        btnhappy = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();
        jLabel1 = new javax.swing.JLabel();
```

```
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setLayout(null);

        Btneven.setBackground(new java.awt.Color(0, 153, 255));
        Btneven.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
        Btneven.setForeground(new java.awt.Color(255, 204, 0));
        Btneven.setText("Even");

Btneven.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelB
order.RAISED));
        Btneven.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                BtnevenActionPerformed(evt);
            }
        });
        jPanel1.add(Btneven);
        Btneven.setBounds(80, 130, 99, 27);

        BtnOdd.setBackground(new java.awt.Color(0, 153, 255));
        BtnOdd.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
        BtnOdd.setForeground(new java.awt.Color(255, 204, 0));
        BtnOdd.setText("Odd");

BtnOdd.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelB
order.RAISED));
        BtnOdd.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                BtnOddActionPerformed(evt);
            }
        });
        jPanel1.add(BtnOdd);
        BtnOdd.setBounds(80, 160, 99, 27);

        btnPrime.setBackground(new java.awt.Color(0, 153, 255));
        btnPrime.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
        btnPrime.setForeground(new java.awt.Color(255, 204, 0));
        btnPrime.setText("Prime");

btnPrime.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.Bevel
Border.RAISED));
        btnPrime.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
            btnPrimeActionPerformed(evt);
        }
    });
    jPanel1.add(btnPrime);
    btnPrime.setBounds(200, 130, 99, 27);

    jButton1.setBackground(new java.awt.Color(0, 153, 255));
    jButton1.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
    jButton1.setForeground(new java.awt.Color(255, 204, 0));
    jButton1.setText("Perfect");

jButton1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelB
order.RAISED));
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });
    jPanel1.add(jButton1);
    jButton1.setBounds(200, 160, 99, 27);

    btnArmstrong.setBackground(new java.awt.Color(0, 153, 255));
    btnArmstrong.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
    btnArmstrong.setForeground(new java.awt.Color(255, 204, 0));
    btnArmstrong.setText("Armstrong");

btnArmstrong.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.B
evelBorder.RAISED));
    btnArmstrong.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnArmstrongActionPerformed(evt);
        }
    });
    jPanel1.add(btnArmstrong);
    btnArmstrong.setBounds(320, 130, 99, 27);

    btnhappy.setBackground(new java.awt.Color(0, 153, 255));
    btnhappy.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
    btnhappy.setForeground(new java.awt.Color(255, 204, 0));
    btnhappy.setText("Happy");

btnhappy.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.Bevel
Border.RAISED));
    btnhappy.addActionListener(new java.awt.event.ActionListener() {
```

```
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnhappyActionPerformed(evt);
            }
        });
        jPanel1.add(btnhappy);
        btnhappy.setBounds(320, 160, 99, 27);

        jLabel3.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
        jLabel3.setForeground(new java.awt.Color(255, 255, 255));
        jLabel3.setText("Finding Numbers");
        jPanel1.add(jLabel3);
        jLabel3.setBounds(150, 80, 210, 29);

        jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/1000_F_78200306_2FBBpWOre0QwjDKB
DZlIo8VZ2L11tkfl.jpg"))); // NOI18N
        jPanel1.add(jLabel1);
        jLabel1.setBounds(-160, 0, 690, 340);

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1,          javax.swing.GroupLayout.DEFAULT_SIZE,          522,
Short.MAX_VALUE)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1,          javax.swing.GroupLayout.DEFAULT_SIZE,          335,
Short.MAX_VALUE)
        );

        pack();
    }// </editor-fold>

    private void BtnevenActionPerformed(java.awt.event.ActionEvent evt) {
      String input = JOptionPane.showInputDialog("Enter number:");
        int n = Integer.parseInt(input);

        if (n % 2 == 0) {
            JOptionPane.showMessageDialog(this,n + " is even");
        } else
        {
            JOptionPane.showMessageDialog(this, n + " is odd");
```

```
    }

}

private void BtnOddActionPerformed(java.awt.event.ActionEvent evt) {
     String input = JOptionPane.showInputDialog("Enter number:");
    int n = Integer.parseInt(input);

    if (n % 2 == 0) {
        JOptionPane.showMessageDialog(this,n + " is even");
    } else
    {
        JOptionPane.showMessageDialog(this, n + " is odd");
    }
}

private void btnPrimeActionPerformed(java.awt.event.ActionEvent evt) {
    String input = JOptionPane.showInputDialog("Enter number:");
int n = Integer.parseInt(input);

if (isPrime(n)) {
    JOptionPane.showMessageDialog(this, n + " is a prime number.");
} else {
    JOptionPane.showMessageDialog(this, n + " is not a prime number.");
}
}
    private boolean isPrime(int num) {
if (num <= 1)
{
    return false;
}

for (int i = 2; i * i <= num; i++) {
    if (num % i == 0)
    {
        return false;
    }
}

return true;

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    String input = JOptionPane.showInputDialog("Enter number:");
int n = Integer.parseInt(input);

if (isPerfect(n)) {
    JOptionPane.showMessageDialog(this, n + " is a perfect number.");
} else {
    JOptionPane.showMessageDialog(this, n + " is not a perfect number.");
}

}
    private boolean isPerfect(int num) {
if (num <= 1)
{
    return false;
}

int sum = 1;
for (int i = 2; i <= num / 2; i++) {
    if (num % i == 0)
    {
        sum += i;
    }
}

return sum == num;



}

private void btnArmstrongActionPerformed(java.awt.event.ActionEvent evt) {

    String input = JOptionPane.showInputDialog("Enter number:");
int n = Integer.parseInt(input);
    if (isArmstrong(n)) {
    JOptionPane.showMessageDialog(this, n + " is a Armstrong number.");
} else {
    JOptionPane.showMessageDialog(this, n + " is not a Armstrong number.");
}

}
  private boolean isArmstrong(int num) {
int oriNumber = num;
int sum = 0;
int numDigits = (int) Math.log10(num) + 1;
```

```
    while (num > 0) {
        int digit = num % 10;
        sum += Math.pow(digit, numDigits);
        num /= 10;
    }

    return sum == oriNumber;

    }

    private void btnhappyActionPerformed(java.awt.event.ActionEvent evt) {
         String input = JOptionPane.showInputDialog("Enter number:");
    int n = Integer.parseInt(input);

    if (isHappy(n)) {
        JOptionPane.showMessageDialog(this, n + " is a happy number.");
    } else {
        JOptionPane.showMessageDialog(this, n + " is not a happy number.");
    }



    }
     private boolean isHappy(int num) {
    HashSet<Integer> Numbers = new HashSet<>();

    while (num != 1 && !Numbers.contains(num)) {
        Numbers.add(num);
        num = calculateSum(num);
    }

    return num == 1;
}

private int calculateSum(int num) {
    int sum = 0;
    while (num > 0) {
        int digit = num % 10;
        sum += digit * digit;
        num /= 10;
    }
    return sum;
```

```
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(Gui.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(Gui.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {
            java.util.logging.Logger.getLogger(Gui.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
            java.util.logging.Logger.getLogger(Gui.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Gui().setVisible(true);
            }
        });
```

```
    }

    // Variables declaration - do not modify
    private javax.swing.JButton BtnOdd;
    private javax.swing.JButton Btneven;
    private javax.swing.JButton btnArmstrong;
    private javax.swing.JButton btnPrime;
    private javax.swing.JButton btnhappy;
    private javax.swing.JButton jButton1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JPanel jPanel1;
    // End of variables declaration
}
```