```java
1 import java.io.IOException;

5
6 /**
7  * This class reads from a input.txt in the same directory as the program, and encrypt it to
   encrypt.txt or decrypt it
8  * to decrypt.txt based on the user's choice.
9  *
10  * @author Minghao Shan
11  * @version 10/4/2017
12  */
13
14 public class CipherDriver {
15     private static Scanner in = new Scanner(System.in);
16     private static Scanner inputReader;
17     private static Scanner encryptReader;
18     private static String output = "";
19
20 /**
21  * This is the encryption method
22  * @throws IOException
23  */
24     public static void encrypt() throws IOException
25     {
26         //Set the Buffered Writer to write on the encrypted file
27         FileWriter fWriter = new FileWriter("encrypt.txt");
28         BufferedWriter bWriter = new BufferedWriter(fWriter);
29
30         while (inputReader.hasNext())
31         {
32             String input = inputReader.nextLine();
33
34             for (int i = 0; i < input.length();i++)
35             {
36                 char inputChar = input.charAt(i);
37                 //encryption
38                 if(inputChar != ' ')
39                 {
40                     inputChar = (char)(inputChar + 3);
41                 }
42                 //add encrypted character to String for output
43                 output = output + inputChar;
44             }
45         }
46         bWriter.write(output);
47         bWriter.close();
48         fWriter.close();
49     }
50     /**
51      * This is the decryption method
52      * @throws IOException
53      */
54     public static void decrypt() throws IOException
55     {
56         //Set the Buffered Writer to write on the decrypted file
57         FileWriter fWriter = new FileWriter("decrypt.txt");
58         BufferedWriter bWriter = new BufferedWriter(fWriter);
59
```

```java
60        while (encryptReader.hasNext())
61        {
62            String input = encryptReader.nextLine();
63            for ( int i = 0; i < input.length() ;i++)
64            {
65                char inputChar = input.charAt(i);
66                //decryption
67                if(inputChar != ' ')
68                {
69                    inputChar = (char) (inputChar - 3);
70                }
71                //add decrypted character to String for output
72                output = output + inputChar;
73            }
74        }
75      bWriter.write(output);
76      bWriter.close();
77      fWriter.close();
78    }
79  /**
80   * method that close the scanner
81   */
82  public static void closeFile()
83  {
84      if(inputReader != null)
85          inputReader.close();
86      if(encryptReader != null)
87          encryptReader.close();
88  }
89  /**
90   *  main program
91   */
92  public static void main(String args[]) {
93
94      try {
95          inputReader = new Scanner (Paths.get("input.txt"));
96          encryptReader = new Scanner (Paths.get("encrypt.txt"));
97      } catch (IOException e1) {
98          System.err.println("Error opening file input.txt or encrypt.txt");
99      }
100
101     System.out.println("Choose 1 for encryption, 2 for decryption");
102     int option = in.nextInt();
103
104     if(option == 1)
105     {
106         try
107         {
108             encrypt();
109             System.out.println("Encryption is finished");
110         }
111         catch ( IOException e)
112         {
113             System.err.println("Error writing encrypted file.");
114         }
115         closeFile();
116     } else if ( option == 2)
```

```
117         {
118             try
119             {
120                 decrypt();
121                 System.out.println("Decryption is finished");
122             }
123             catch (IOException e)
124             {
125                 System.err.println("Error writing deprypted file.");
126             }
127             closeFile();
128         } else {
129             System.out.println("Wrong entry \n");
130         }
131     }
132 }
133
```