 ****

**实验报告三**

学生姓名： 陈华豪 学 号： 6130116238 专业班级： 16级网络工程

实验类型：√验证 □ 综合 □设计□创新 实验日期： 2018.12.22 实验成绩：

1. 实验项目名称

RSA算法的实现。

1. 实验目的

RSA公开密钥密码体制。所谓的公开密钥密码体制就是使用不同的加密密钥与解密密钥，是一种“由已知加密密钥推导出解密密钥在计算上是不可行的”密码体制。

通过RSA算法s实现对图片与文本两种内容进行加解密。

1. 实验基本原理

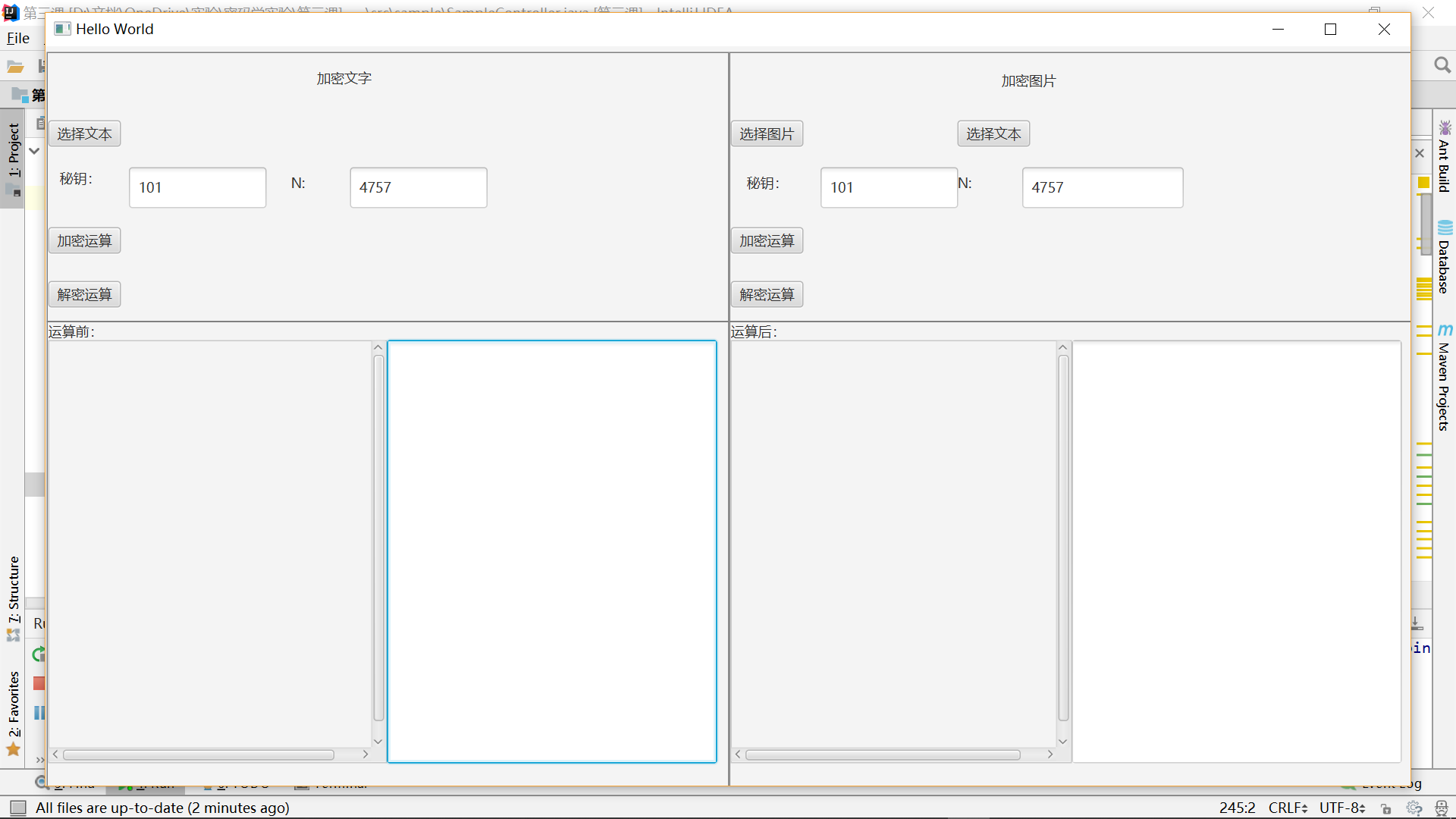
RSA的安全性依赖于大数分解，但是否等同于大数分解一直未能得到理论上的证明，因为没有证明破解RSA就一定需要作大数分解。假设存在一种无须分解大数的算法，那它肯定可以修改成为大数分解算法。 RSA 的一些变种算法已被证明等价于大数分解。不管怎样，分解n是最显然的攻击方法。因此，模数n必须选大一些，因具体适用情况而定。

实验基本原理是将加密对象用一定规则进行编码，然后按照相应的方法对编码后的对象分组转换为数字进行RSA的加密，然后对应进行解密。

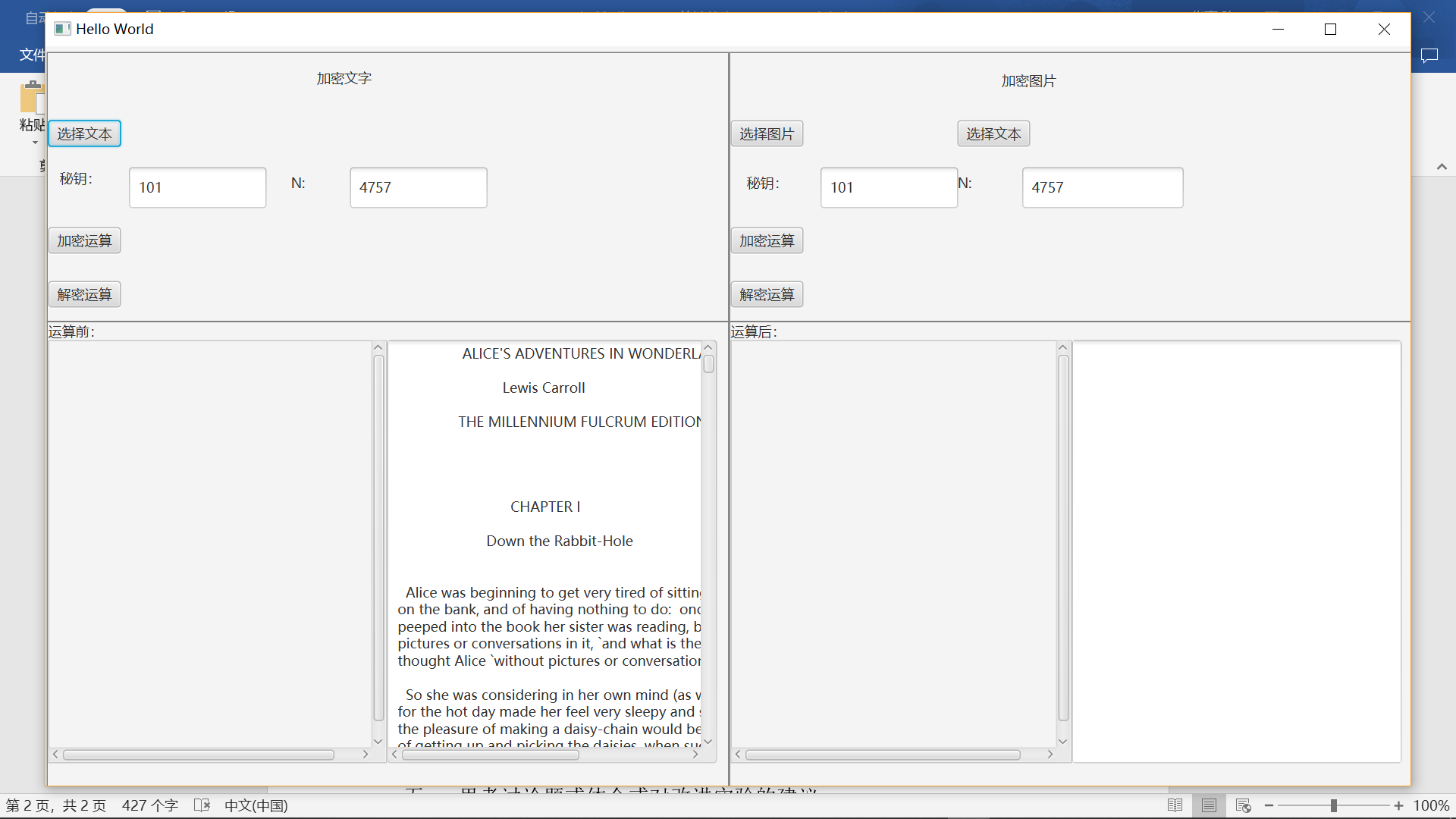
1. 实验数据及处理结果

默认公钥（e=101，N=4757），私钥（d=1601,N=4757）；

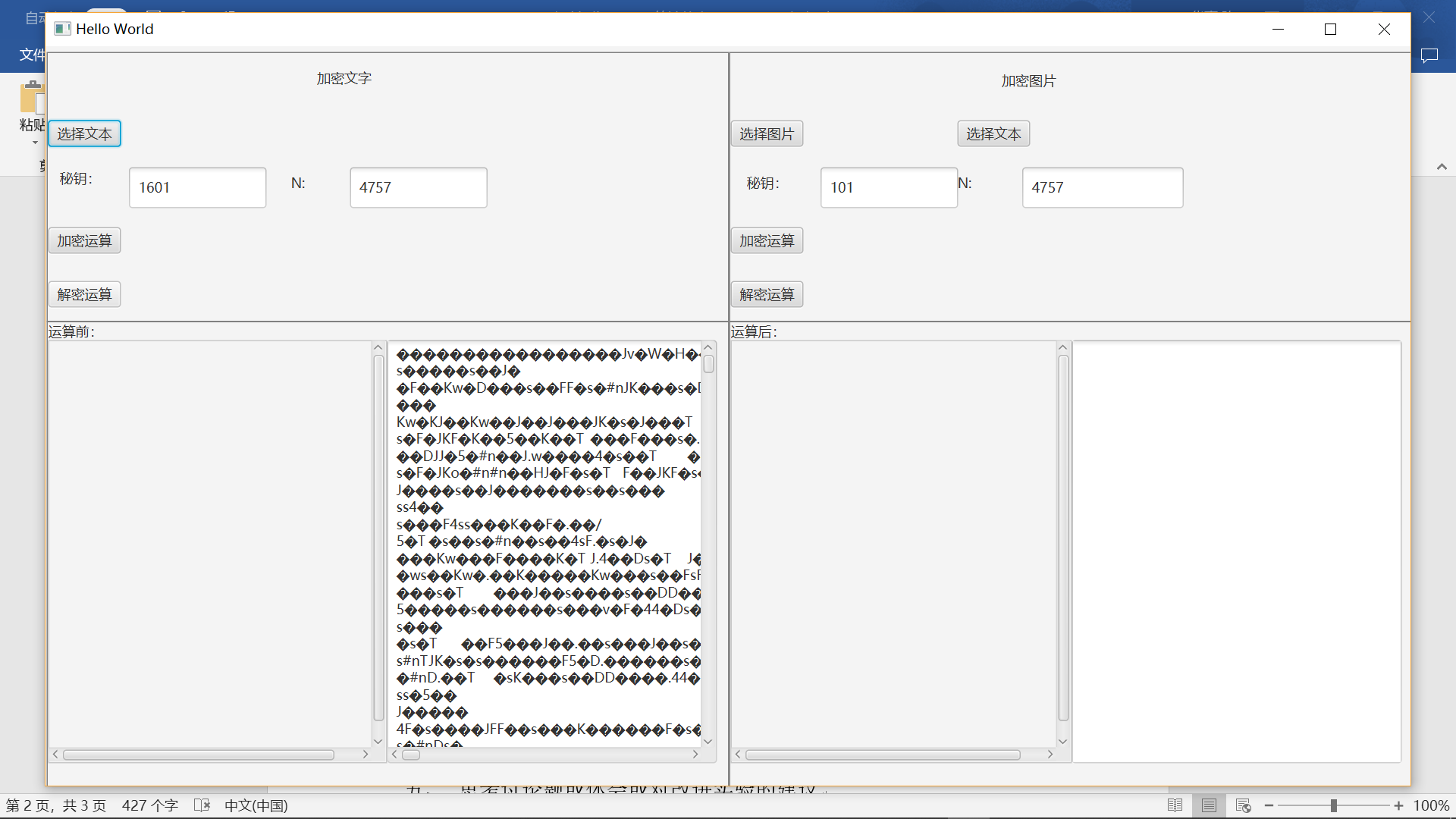
开始窗口



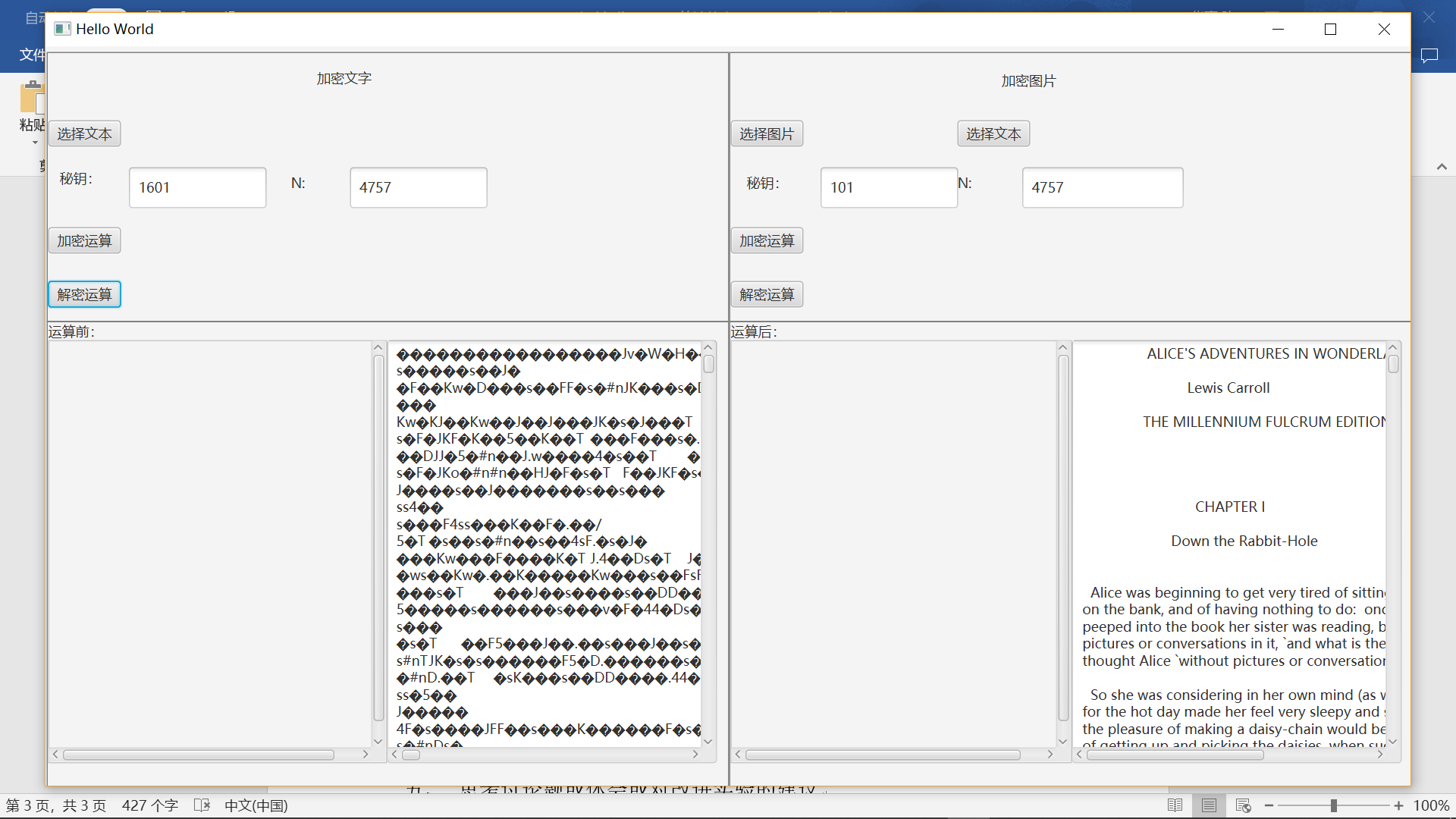
选择明文文本并显示在左侧, 更改秘钥为公钥e=101



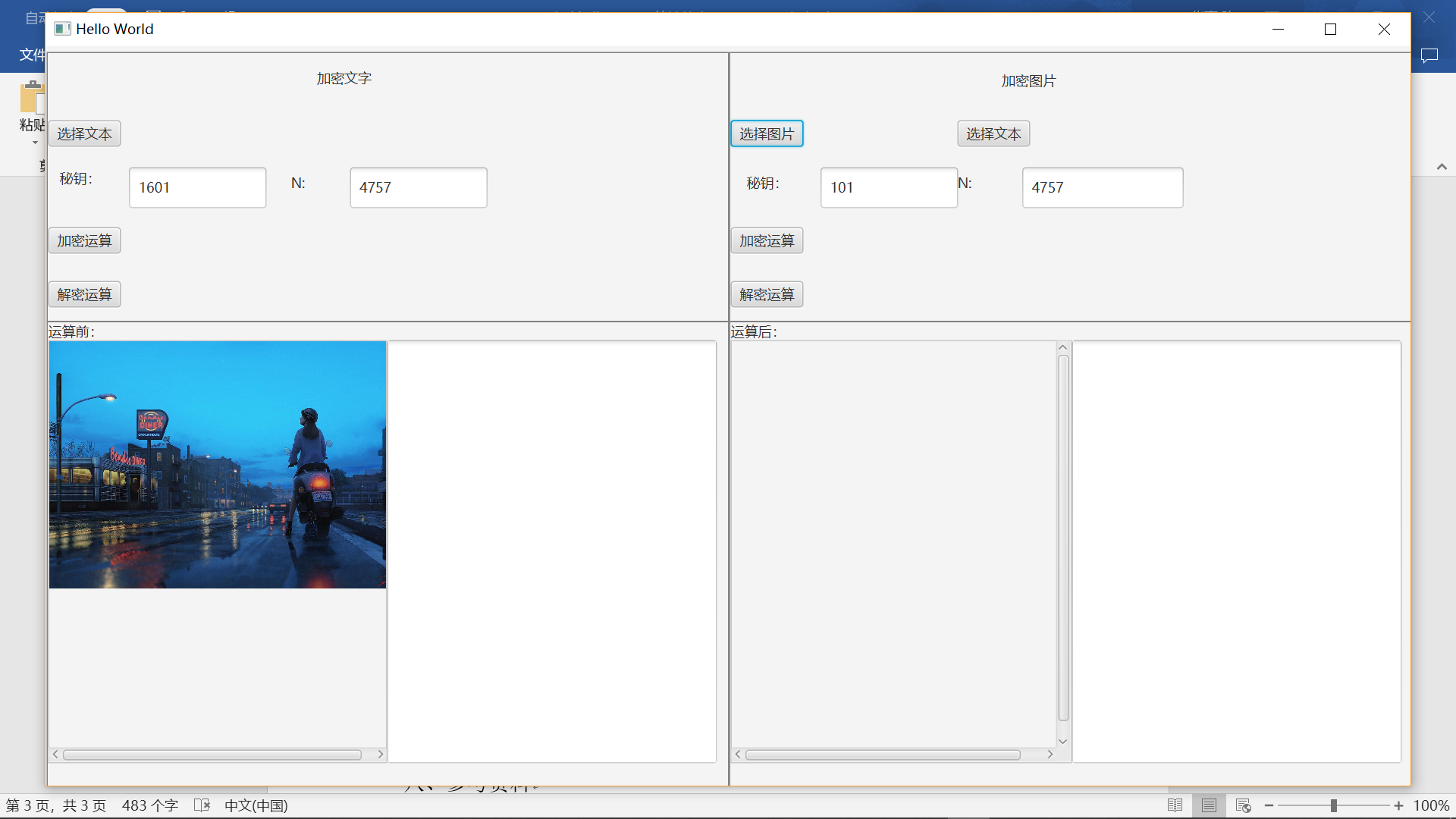
选择所选加密后的密文文本并显示在左侧，更改秘钥为私钥d=1601



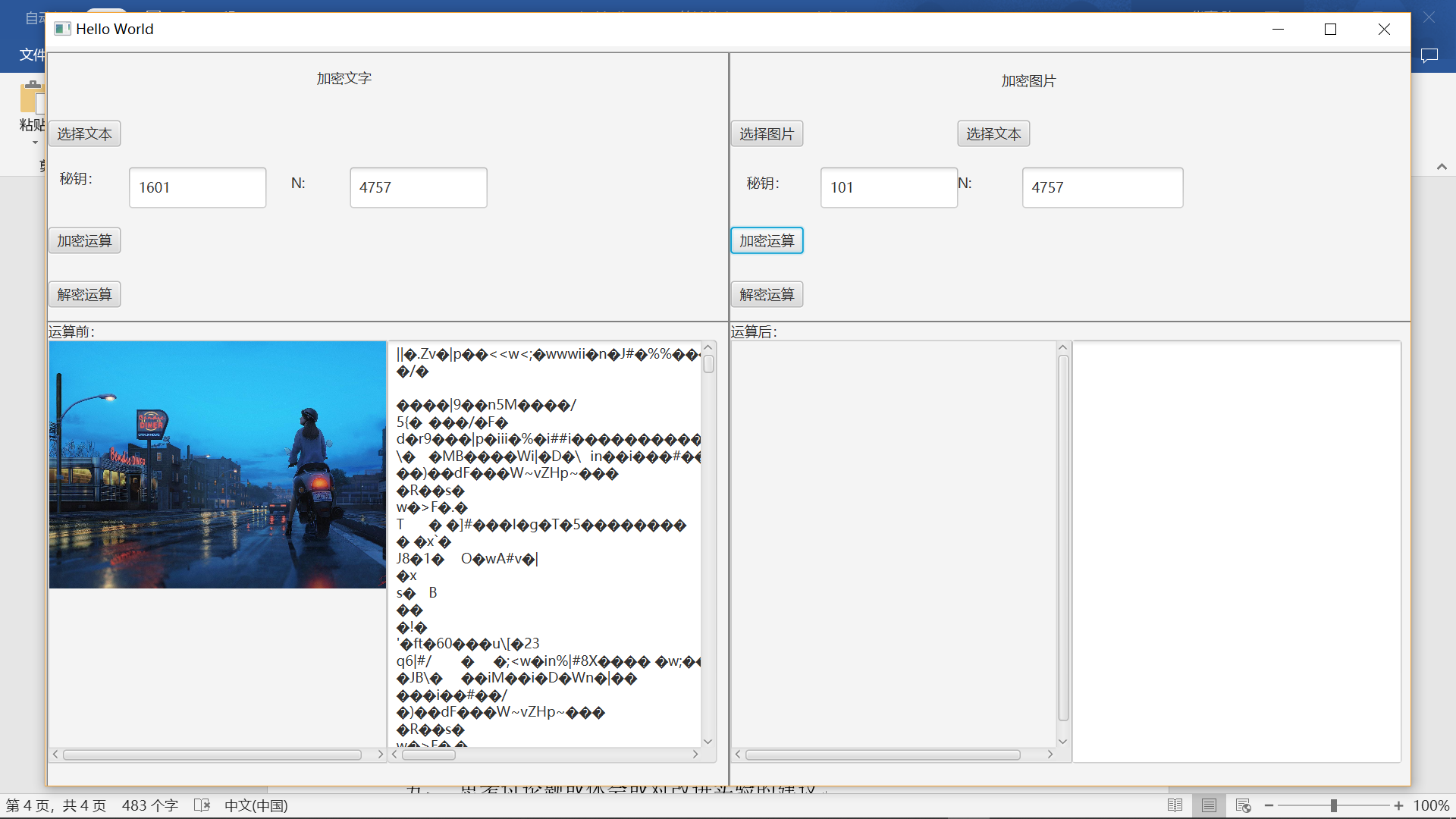
点击“解密运算”，解密密文并将其显示在右侧。



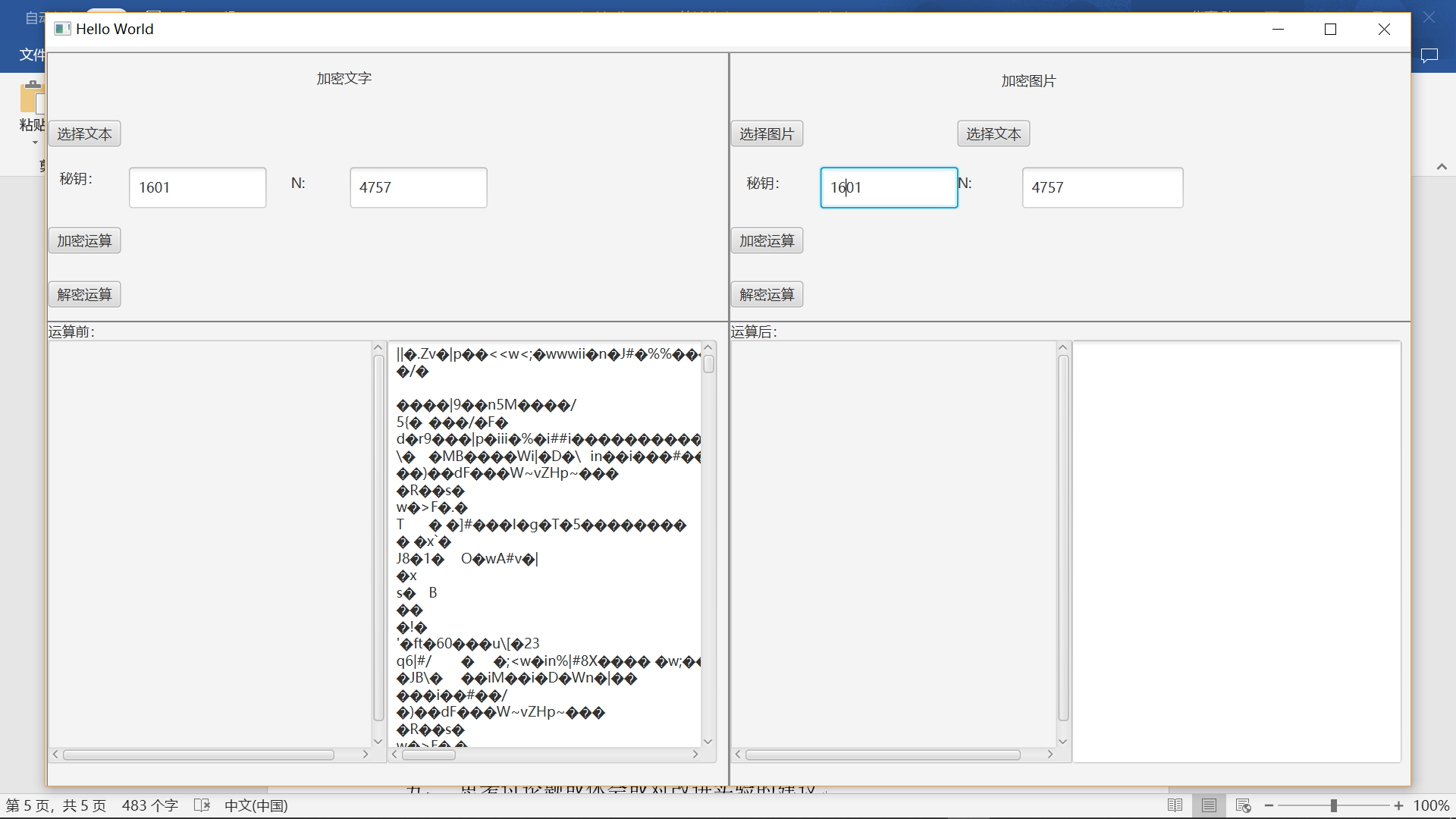
选择要加密的图片并显示在左侧，更改秘钥为公钥e=101



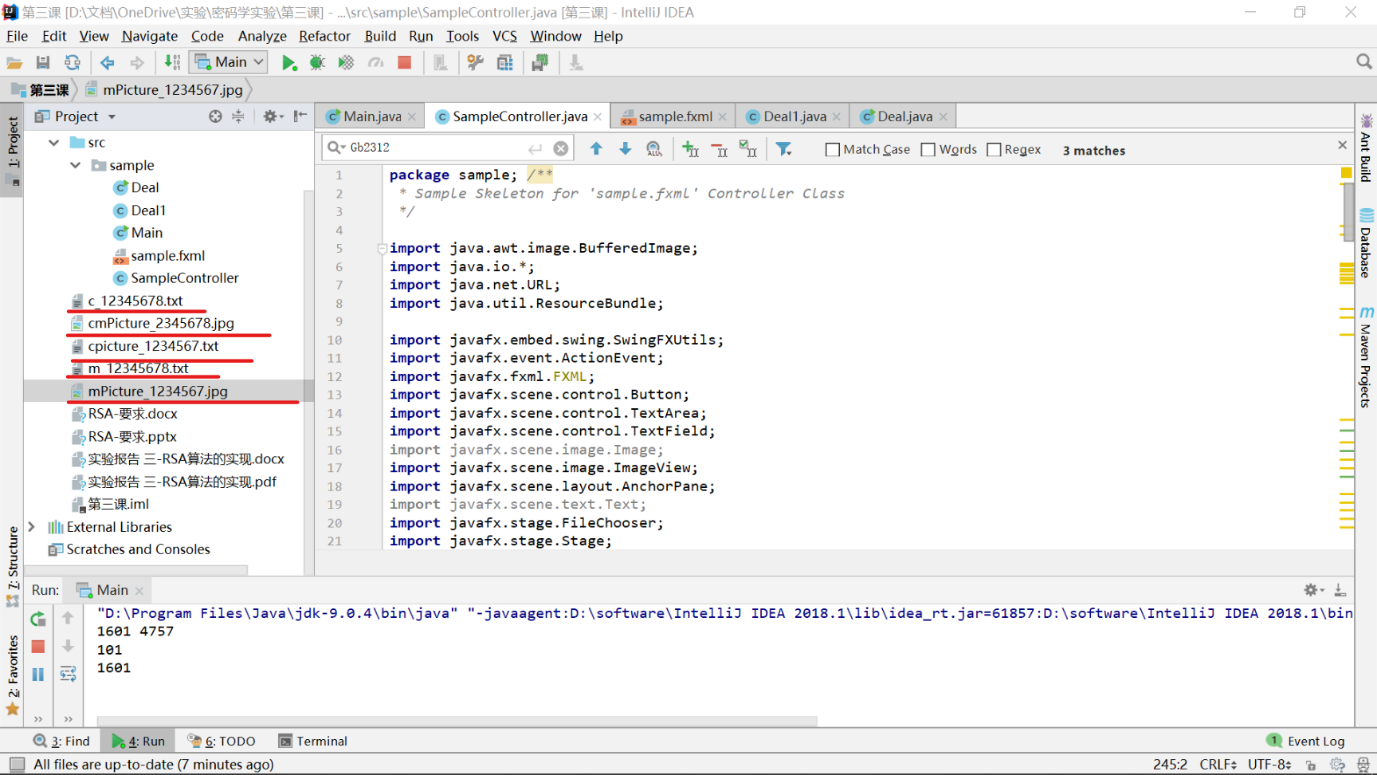
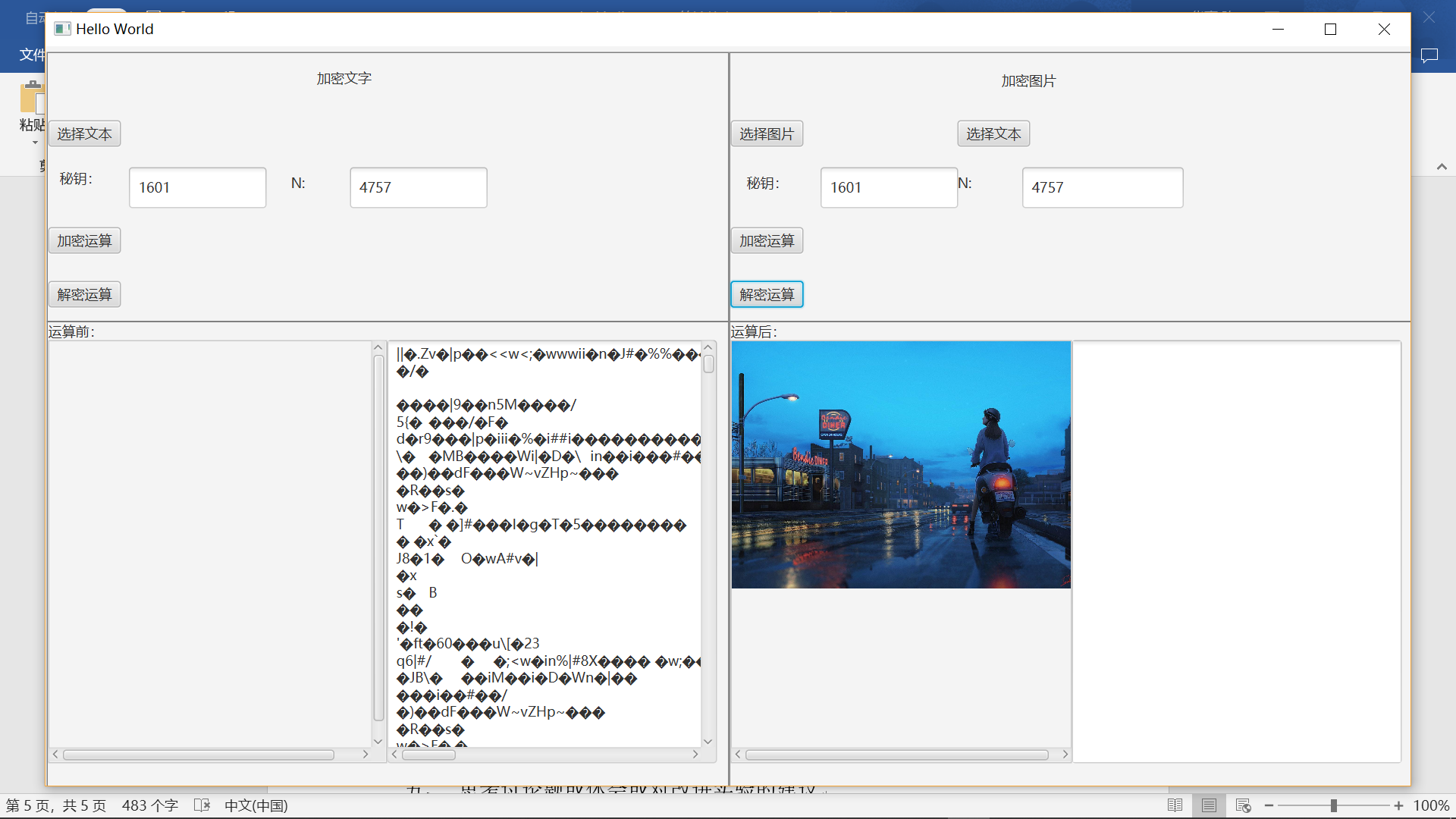
点击加密运算并将加密后的密文文本显示在图片右边



选择加密后的密文文本并显示在左侧，更改秘钥为私钥d=1601



点击解密运算，解密密文文本，并将解密后的道德图片显示在右侧



代码：Controller类:

SampleController.java:

1. **package** sample; /\*\*
2. \* Sample Skeleton for 'sample.fxml' Controller Class
3. \*/
5. **import** java.awt.image.BufferedImage;
6. **import** java.io.\*;
7. **import** java.net.URL;
8. **import** java.util.ResourceBundle;
10. **import** javafx.embed.swing.SwingFXUtils;
11. **import** javafx.event.ActionEvent;
12. **import** javafx.fxml.FXML;
13. **import** javafx.scene.control.Button;
14. **import** javafx.scene.control.TextArea;
15. **import** javafx.scene.control.TextField;
16. **import** javafx.scene.image.Image;
17. **import** javafx.scene.image.ImageView;
18. **import** javafx.scene.layout.AnchorPane;
19. **import** javafx.scene.text.Text;
20. **import** javafx.stage.FileChooser;
21. **import** javafx.stage.Stage;
23. **import** javax.imageio.ImageIO;
25. **public** **class** SampleController {
27. **public** TextField n2;
28. **public** TextField n1;
29. **byte**[] image1;//字节型数组存储选择的图片
30. File image2;//File类型存储解密后的图片
31. **byte**[] txtone;//字节型数组存储选择的文本
32. **byte**[] txttwo;//字节型数组存储加密后的txtone；
33. **byte**[] txtcipher;//字节型数组存储加密后的文本
34. **int** N=4757;
35. **int** d=1601;
36. **int** e=101;
38. @FXML
39. **private** TextArea txt2;
41. @FXML
42. **private** TextArea txt1;
44. @FXML // ResourceBundle that was given to the FXMLLoader
45. **private** ResourceBundle resources;
47. @FXML // URL location of the FXML file that was given to the FXMLLoader
48. **private** URL location;
50. @FXML // fx:id="txt"
51. **private** Button txt; // Value injected by FXMLLoader
53. @FXML // fx:id="controller"
54. **private** AnchorPane controller; // Value injected by FXMLLoader
56. @FXML // fx:id="img"
57. **private** Button img; // Value injected by FXMLLoader
59. @FXML // fx:id="encrype1"
60. **private** Button encrype1; // Value injected by FXMLLoader
62. @FXML // fx:id="decrype1"
63. **private** Button decrype1; // Value injected by FXMLLoader
65. @FXML // fx:id="secret2"
66. **private** TextField secret2; // Value injected by FXMLLoader
68. @FXML // fx:id="decrype2"
69. **private** Button decrype2; // Value injected by FXMLLoader
71. @FXML // fx:id="img2"
72. **private** ImageView img2; // Value injected by FXMLLoader
74. @FXML // fx:id="img1"
75. **private** ImageView img1; // Value injected by FXMLLoader
77. @FXML // fx:id="secret1"
78. **private** TextField secret1; // Value injected by FXMLLoader
80. @FXML // fx:id="encrype2"
81. **private** Button encrype2; // Value injected by FXMLLoader
83. @FXML   //选择文本并显示
84. **void** chooseTxt(ActionEvent event) **throws** UnsupportedEncodingException {
85. File file;
86. file = fileChoose();
87. fileToBytes(file,**true**);
88. txt1.setText(**new** String(txtone,"GB2312"));
89. }
91. @FXML   //选择图片并显示
92. **void** chooseImage(ActionEvent event) **throws** Exception {
93. File file;
94. file = fileChoose();
95. showImg(file,**false**);
96. }
98. @FXML   //解密文本并显示
99. **void** decrypeTxt(ActionEvent event) **throws** UnsupportedEncodingException {
100. **this**.d=Integer.parseInt(secret1.getText());
101. **this**.N=Integer.parseInt(n1.getText());
102. System.out.println(d+ " "+ N);
103. txttwo=Deal.decrype(txtone,d,N);//转译文本加密为字节型数组

106. txt2.setText(**new** String(txttwo,"GB2312"));//显示密文
107. }
109. @FXML   //加密文本并显示
110. **void** encrypeTxt(ActionEvent event) **throws** UnsupportedEncodingException {
111. **this**.e=Integer.parseInt(secret1.getText());
112. **this**.N=Integer.parseInt(n1.getText());
113. System.out.println(e + " " + N);
114. txttwo=Deal.encrype(txtone,e,N);//转译文本加密为字节型数组
116. }
118. @FXML   //解密图片并显示
119. **void** decrypeImage(ActionEvent event) **throws** Exception {
120. **byte**[] image;//存储转译后的二进制密文
121. **this**.d=Integer.parseInt(secret2.getText());
122. **this**.N=Integer.parseInt(n2.getText());
123. System.out.println(d);
124. image=Deal.decrype(txtone,d,N);
125. bytesToImgFile(image);
126. showImg(image2,**true**);
127. }
129. @FXML   //加密图片
130. **void** encrypeImage(ActionEvent event) **throws** Exception {
131. **byte**[] image;//存储转译后的二进制密文
132. **this**.e=Integer.parseInt(secret2.getText());
133. **this**.N=Integer.parseInt(n2.getText());
134. image=Deal.encrype(image1,e,N);
135. System.out.println(e);
136. String pathname ="D:\\文档\\OneDrive\\实验\\密码学实验\\第三课\\cpicture\_1234567.txt";
137. bytesToFile(image,pathname);
138. File file = **new** File(pathname);
139. fileToBytes(file,**false**);
140. txt1.setText(**new** String(txtcipher,"GB2312"));//显示密文
141. }
143. @FXML // This method is called by the FXMLLoader when initialization is complete
144. **void** initialize() {
145. **assert** txt != **null** : "fx:id=\"txt\" was not injected: check your FXML file 'sample.fxml'.";
146. **assert** controller != **null** : "fx:id=\"controller\" was not injected: check your FXML file 'sample.fxml'.";
147. **assert** img != **null** : "fx:id=\"img\" was not injected: check your FXML file 'sample.fxml'.";
148. **assert** encrype1 != **null** : "fx:id=\"encrype1\" was not injected: check your FXML file 'sample.fxml'.";
149. **assert** decrype1 != **null** : "fx:id=\"decrype1\" was not injected: check your FXML file 'sample.fxml'.";
150. **assert** secret2 != **null** : "fx:id=\"secret2\" was not injected: check your FXML file 'sample.fxml'.";
151. **assert** decrype2 != **null** : "fx:id=\"decrype2\" was not injected: check your FXML file 'sample.fxml'.";
152. **assert** img2 != **null** : "fx:id=\"img2\" was not injected: check your FXML file 'sample.fxml'.";
153. **assert** img1 != **null** : "fx:id=\"img1\" was not injected: check your FXML file 'sample.fxml'.";
154. **assert** secret1 != **null** : "fx:id=\"secret1\" was not injected: check your FXML file 'sample.fxml'.";
155. **assert** encrype2 != **null** : "fx:id=\"encrype2\" was not injected: check your FXML file 'sample.fxml'.";
157. }
159. //文件选择器，选择时先进行清空所有文本图片域操作。
160. File fileChoose(){
161. txt1.clear();
162. txt2.clear();
163. img1.setImage(**null**);
164. img2.setImage(**null**);
165. FileChooser fileChooser = **new** FileChooser();
166. fileChooser.setTitle("Open Resource File");
167. fileChooser.setInitialDirectory(**new** File(System.getProperty("user.dir")));
168. File file = fileChooser.showOpenDialog(**new** Stage());
169. **return** file;
170. }
172. //显示图片，若b为true则显示在“img2”,否则显示在“img1”
173. **void** showImg(File file,**boolean** b) **throws** Exception{
175. ByteArrayOutputStream baos = **new** ByteArrayOutputStream();
176. **try** {
178. BufferedImage bi;
179. bi = ImageIO.read(file);
180. **if**(b){
181. img2.setImage(SwingFXUtils.toFXImage(bi, **null**));
182. }**else**{
183. img1.setImage(SwingFXUtils.toFXImage(bi, **null**));
184. ImageIO.write(bi, "jpg", baos);
185. **this**.image1 = baos.toByteArray();
187. }
188. //            System.err.println(image1.length);
189. } **catch** (IOException e) {
190. e.printStackTrace();
191. }**catch** (Exception e) {
192. e.printStackTrace();
193. }**finally** {
194. baos.close();
195. }
196. }
198. //将字节型数组转化成File型，传给image2
199. **void** bytesToImgFile(**byte**[] image1) **throws** Exception {
200. ByteArrayInputStream bais = **new** ByteArrayInputStream(image1);
201. BufferedImage bi1 =ImageIO.read(bais);
202. **try** {
203. File w2 = **new** File("D:\\文档\\OneDrive\\实验\\密码学实验\\第三课\\cmPicture\_2345678.jpg");
204. ImageIO.write(bi1, "jpg", w2);
205. **this**.image2 = w2;
206. } **catch** (IOException e) {
207. e.printStackTrace();
208. }**finally**{
209. bais.close();
210. }
211. }
213. //将File类型转化成byte数组，若a为真，传给txtone,否则传给txtcipher
214. **void** fileToBytes(File file,**boolean** a){
216. **try** {
217. FileInputStream in=**new** FileInputStream(file);
218. // size  为字串的长度 ，这里一次性读完
219. **int** size=in.available();
220. **byte**[] buffer=**new** **byte**[size];
221. in.read(buffer);
222. in.close();
223. **if**(a){
224. **this**.txtone=buffer;
225. }**else**{
226. **this**.txtcipher=buffer;
227. }
229. } **catch** (IOException e) {
230. e.printStackTrace();
231. }
232. }
234. //将byte数组转化成file类型txt文档，保存在pathname。
235. **public** **void** bytesToFile(**byte**[] bytes,String pathname) {
236. **try** {
237. FileOutputStream fos = **new** FileOutputStream(pathname);
238. fos.write(bytes);
239. fos.close();
240. } **catch** (Exception e) {
241. // TODO Auto-generated catch block
242. e.printStackTrace();
243. }
244. }
245. }

Main.java:

1. **package** sample;
3. **import** javafx.application.Application;
4. **import** javafx.fxml.FXMLLoader;
5. **import** javafx.scene.Parent;
6. **import** javafx.scene.Scene;
7. **import** javafx.stage.Stage;
9. **public** **class** Main **extends** Application {


13. @Override
14. **public** **void** start(Stage primaryStage) **throws** Exception{
15. Parent root = FXMLLoader.load(getClass().getResource("sample.fxml"));
17. primaryStage.setTitle("Hello World");
18. primaryStage.setScene(**new** Scene(root, 1200, 650));
19. primaryStage.show();
20. }

23. **public** **static** **void** main(String[] args) {
24. launch(args);
25. }
26. }

Deal.java:

1. **package** sample;
3. /\*\*
4. \* @Author: Chen Huahao
5. \* @Dscription:
6. \* @Date: Created in 19:57 2018/12/20
7. \*/
8. **public** **class** Deal {
9. **public** **static** **byte**[] decrype(**byte**[] byte5, **int** e, **int** N) {
10. **int**[] bytes=read\_2timesbytes(byte5);
11. //        for (int i = 0; i <bytes.length ; i++) {
12. //            System.out.println(bytes[i]);;
13. //        }
15. **int** rr[] =read\_2timesbytes(byte5);
16. String m= Integer.toBinaryString(e);
18. **int**[] mresult = **new** **int**[m.length()];
20. **for** (**int** i = 0; i <mresult.length; i++) {
21. **char** c = m.charAt(mresult.length-i-1);
22. mresult[i] = Character.getNumericValue(c);
23. }

26. **int**[] x = **new** **int**[rr.length];
27. **for** (**int** i = 0; i < rr.length; i++) {
29. x[i]=1;
30. **for** (**int** j = m.length()-1; j >=0; j--) {
31. x[i]=(x[i]\*x[i])%N;
32. //                System.out.println(mresult[j]);
33. **if**(mresult[j]==1)
34. x[i]=(x[i]\*rr[i])%N;
36. }
37. //            System.out.println(x[i]+"bb");
39. }
40. **byte**[] bytes2 = **new** **byte**[bytes.length];
42. **for** (**int** i = 0; i < bytes.length; i++) {
43. bytes2[i]=(**byte**)x[i];
44. }
45. **return** bytes2;
46. }

49. **public** **static** **byte**[] bytetimes2(**int**[] x) {
50. **byte**[] bytes2= **new** **byte**[x.length\*2];
51. **for** (**int** i = 0; i < x.length; i++) {
52. bytes2[i\*2]=(**byte**)(x[i]);
53. bytes2[i\*2+1]=(**byte**)(x[i]>>8);
54. }
55. **return** bytes2;
56. }
58. **public** **static** **byte**[] encrype(**byte**[] bytes,**int** e,**int** N){
60. String m= Integer.toBinaryString(e);
62. **int**[] mresult = **new** **int**[m.length()];
64. **for** (**int** i = 0; i <mresult.length; i++) {
65. **char** c = m.charAt(mresult.length-i-1);
66. mresult[i] = Character.getNumericValue(c);
67. }
69. **int**[] rr = **new** **int**[bytes.length] ;
70. **int**[] x = **new** **int**[bytes.length];
71. **for** (**int** i = 0; i < bytes.length; i++) {
72. rr[i]= bytes[i] & 0xFF;
74. x[i]=1;
75. **for** (**int** j = m.length()-1; j >=0; j--) {
76. x[i]=(x[i]\*x[i])%N;
77. //                System.out.println(mresult[j]);
78. **if**(mresult[j]==1)
79. x[i]=(x[i]\*rr[i])%N;
81. }
82. //            System.out.println(x[i]+"bb");
84. }
86. //        byte[] bytes2 = new byte[x.length];
87. //
88. //        for (int i = 0; i < x.length; i++) {
89. //            bytes2[i]=(byte)(x[i]);
90. //            byte by = (byte)4016;
91. //            System.out.println(by & 0xff);
92. //        }
93. **return** bytetimes2(x);
95. }
97. **public** **static** **int**[] read\_2timesbytes(**byte**[] bytes){
98. **int**[] x=**new** **int**[bytes.length/2];
99. **for** (**int** i = 0; i <x.length; i++) {
100. **int** t=bytes[2\*i]&0xFF;
101. t=(t<0?t+256:t);
102. t+=(bytes[2\*i+1]&0xFF)\*256;
103. x[i]=t;
104. }
105. **return** x;
106. }

109. **public** **static** **void** main(String[] args) {
110. **byte**[] bytes={1,2,3,4,5};
111. **byte**[] bytes1=encrype(bytes,101,4757);
112. **byte**[] bytes2=decrype(bytes1,1601,4757);
113. **for** (**int** i = 0; i <bytes2.length ; i++) {
114. System.out.println(bytes2[i]);
115. }
116. }
117. }

Fxml:

1. **<?xml** version="1.0" encoding="UTF-8"**?>**
3. **<?import** javafx.scene.text.\***?>**
4. **<?import** javafx.scene.image.\***?>**
5. **<?import** java.lang.\***?>**
6. **<?import** java.util.\***?>**
7. **<?import** javafx.scene.\***?>**
8. **<?import** javafx.scene.control.\***?>**
9. **<?import** javafx.scene.layout.\***?>**
11. **<AnchorPane** prefHeight="650.0" prefWidth="1200.0" xmlns="http://javafx.com/javafx/8" xmlns:fx="http://javafx.com/fxml/1" fx:controller="sample.SampleController"**>**
12. **<children>**
13. **<GridPane** layoutX="1.0" layoutY="5.0" prefHeight="650.0" prefWidth="1200.0"**>**
14. **<columnConstraints>**
15. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
16. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
17. **</columnConstraints>**
18. **<rowConstraints>**
19. **<RowConstraints** maxHeight="383.0" minHeight="10.0" prefHeight="312.0" vgrow="SOMETIMES" **/>**
20. **<RowConstraints** maxHeight="499.0" minHeight="10.0" prefHeight="488.0" vgrow="SOMETIMES" **/>**
21. **</rowConstraints>**
22. **<children>**
23. **<VBox** prefHeight="200.0" prefWidth="100.0" style="-fx-border-color: grey" GridPane.rowIndex="1"**>**
24. **<children>**
25. **<Label** text="运算前：" **/>**
26. **<HBox** prefHeight="372.0" prefWidth="598.0"**>**
27. **<children>**
28. **<ScrollPane** prefHeight="372.0" prefWidth="298.0"**>**
29. **<content>**
30. **<ImageView** fx:id="img1" fitHeight="369.0" fitWidth="307.0" pickOnBounds="true" preserveRatio="true" **/>**
31. **</content>**
32. **</ScrollPane>**
33. **<TextArea** fx:id="txt1" prefHeight="372.0" prefWidth="290.0" **/>**
34. **</children>**
35. **</HBox>**
36. **</children>**
37. **</VBox>**
38. **<VBox** prefHeight="200.0" prefWidth="100.0" style="-fx-border-color: grey" GridPane.columnIndex="1" GridPane.rowIndex="1"**>**
39. **<children>**
40. **<Label** text="运算后：" **/>**
41. **<HBox** prefHeight="372.0" prefWidth="598.0"**>**
42. **<children>**
43. **<ScrollPane** prefHeight="372.0" prefWidth="300.0"**>**
44. **<content>**
45. **<ImageView** fx:id="img2" fitHeight="369.0" fitWidth="307.0" pickOnBounds="true" preserveRatio="true" **/>**
46. **</content>**
47. **</ScrollPane>**
48. **<TextArea** fx:id="txt2" prefHeight="372.0" prefWidth="290.0" **/>**
49. **</children>**
50. **</HBox>**
51. **</children>**
52. **</VBox>**
53. **<GridPane** style="-fx-border-color: grey"**>**
54. **<columnConstraints>**
55. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
56. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
57. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
58. **</columnConstraints>**
59. **<rowConstraints>**
60. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
61. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
62. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
63. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
64. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
65. **</rowConstraints>**
66. **<children>**
67. **<Button** fx:id="decrype1" mnemonicParsing="false" onAction="#decrypeTxt" text="解密运算" GridPane.rowIndex="4" **/>**
68. **<Button** fx:id="encrype1" mnemonicParsing="false" onAction="#encrypeTxt" text="加密运算" GridPane.rowIndex="3" **/>**
69. **<Button** fx:id="txt" mnemonicParsing="false" onAction="#chooseTxt" text="选择文本" GridPane.rowIndex="1" **/>**
70. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.columnIndex="1"**>**
71. **<children>**
72. **<Label** layoutX="37.0" layoutY="14.0" text="加密文字" **/>**
73. **</children>**
74. **</Pane>**
75. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.rowIndex="2"**>**
76. **<children>**
77. **<Label** layoutX="10.0" layoutY="8.0" text="秘钥：" **/>**
78. **<TextField** fx:id="secret1" layoutX="71.0" layoutY="6.0" prefHeight="36.0" prefWidth="121.0" text="101" **/>**
79. **</children>**
80. **</Pane>**
81. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.columnIndex="1" GridPane.rowIndex="2"**>**
82. **<children>**
83. **<Label** layoutX="14.0" layoutY="12.0" text="N:" **/>**
84. **<TextField** fx:id="n1" layoutX="66.0" layoutY="6.0" prefHeight="36.0" prefWidth="121.0" text="4757" **/>**
85. **</children>**
86. **</Pane>**
87. **</children>**
88. **</GridPane>**
89. **<GridPane** style="-fx-border-color: grey" GridPane.columnIndex="1"**>**
90. **<children>**
91. **<Label** GridPane.rowIndex="1" **/>**
92. **<Button** fx:id="decrype2" mnemonicParsing="false" onAction="#decrypeImage" text="解密运算" GridPane.rowIndex="4" **/>**
93. **<Button** fx:id="encrype2" mnemonicParsing="false" onAction="#encrypeImage" text="加密运算" GridPane.rowIndex="3" **/>**
94. **<Button** fx:id="img" mnemonicParsing="false" onAction="#chooseImage" text="选择图片" GridPane.rowIndex="1" **/>**
95. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.columnIndex="1"**>**
96. **<children>**
97. **<Label** layoutX="39.0" layoutY="16.0" text="加密图片" **/>**
98. **</children>**
99. **</Pane>**
100. **<Button** mnemonicParsing="false" onAction="#chooseTxt" text="选择文本" GridPane.columnIndex="1" GridPane.rowIndex="1" **/>**
101. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.rowIndex="2"**>**
102. **<children>**
103. **<Label** layoutX="14.0" layoutY="12.0" text="秘钥：" **/>**
104. **<TextField** fx:id="secret2" layoutX="79.0" layoutY="6.0" prefHeight="36.0" prefWidth="121.0" text="101" **/>**
105. **</children>**
106. **</Pane>**
107. **<Pane** prefHeight="200.0" prefWidth="200.0" GridPane.columnIndex="1" GridPane.rowIndex="2"**>**
108. **<children>**
109. **<Label** layoutY="12.0" text="N:" **/>**
110. **<TextField** fx:id="n2" layoutX="57.0" layoutY="6.0" prefHeight="36.0" prefWidth="142.0" text="4757" **/>**
111. **</children>**
112. **</Pane>**
113. **</children>**
114. **<columnConstraints>**
115. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
116. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
117. **<ColumnConstraints** hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" **/>**
118. **</columnConstraints>**
119. **<rowConstraints>**
120. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
121. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
122. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
123. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
124. **<RowConstraints** minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" **/>**
125. **</rowConstraints>**
126. **</GridPane>**
127. **</children>**
128. **</GridPane>**
129. **</children>**
130. **</AnchorPane>**