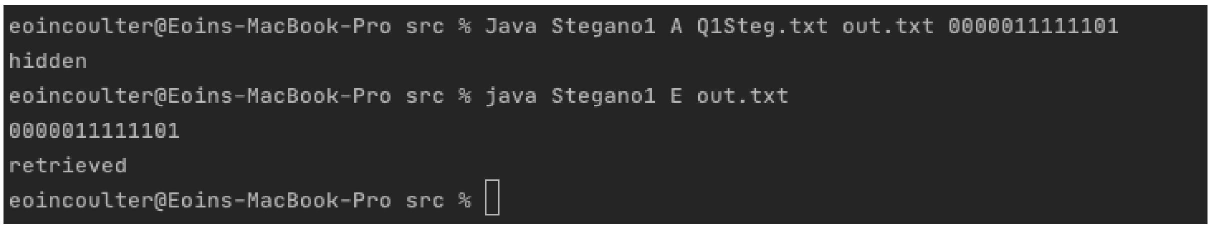
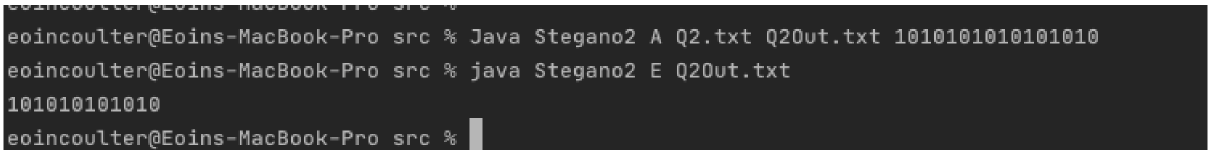
Q1 Screen Shot



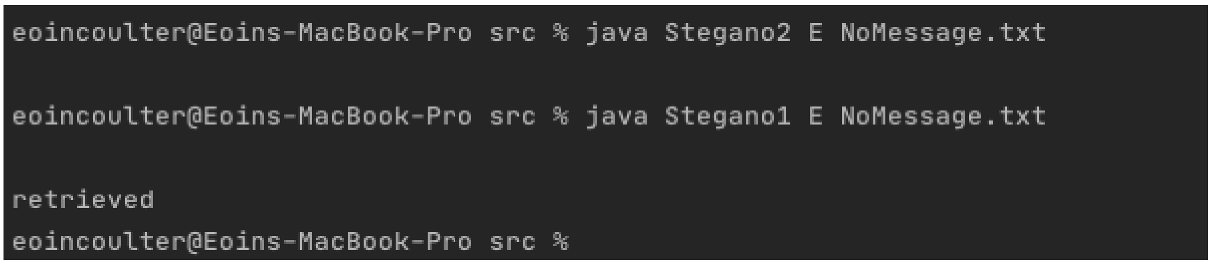
```
eoincoulter@Eoins-MacBook-Pro src % Java Stegano1 A Q1Steg.txt out.txt 0000011111101
hidden
eoincoulter@Eoins-MacBook-Pro src % java Stegano1 E out.txt
0000011111101
retrieved
eoincoulter@Eoins-MacBook-Pro src % 
```

Q2 Screen Shot



```
eoincoulter@Eoins-MacBook-Pro src % Java Stegano2 A Q2.txt Q2Out.txt 1010101010101010
eoincoulter@Eoins-MacBook-Pro src % java Stegano2 E Q2Out.txt
101010101010
eoincoulter@Eoins-MacBook-Pro src % 
```

Screen Shot showing nothing retrieved from normal file



```
eoincoulter@Eoins-MacBook-Pro src % java Stegano2 E NoMessage.txt

eoincoulter@Eoins-MacBook-Pro src % java Stegano1 E NoMessage.txt

retrieved
eoincoulter@Eoins-MacBook-Pro src %
```

CODE FOR STEGANO1

```java
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class Stegano1
{
    /**
     * Constructor for objects of class Stegano1
     */
    public Stegano1()
    {
    }

    public static void main(String[] args) {
        String arg1, arg2, arg3, arg4;
        Boolean err = false;

        if (args != null && args.length > 1) { // Check for minimum number
of arguments
            arg1 = args[0];
            arg2 = args[1];

            if (arg2 == "") {
                err = true;
            }
            else if ((arg1.equals( "A")) && (args.length > 3)){
                // Get other arguments
                arg3 = args[2];
                arg4 = args[3];
                if (arg3 == "" || arg4 == "") {
                    err = true;
                }
                else {
                    // Hide bitstring
                    hide(arg2, arg3, arg4);
                }
            }
            else if (arg1.equals("E")){
                // Extract bitstring from text
                retrieve(arg2);
            }
            else {
                err = true;
            }
        }
        else {
            err = true;
        }

        if (err == true) {
            System.out.println();
            System.out.println("    Stegano1 <A:E><Input
File><OutputFile><Bitstring>");
            System.out.println("Example: Stegano1 A inp.txt out.txt
0010101");
            System.out.println("Example: Stegano1 E inp.txt");

        }
    }

    static void hide(String inpFile, String outFile, String binString) {
        //
        BufferedReader reader;
        BufferedWriter writer;

        try {
            reader = new BufferedReader(new FileReader(inpFile));
            writer = new BufferedWriter(new FileWriter(outFile));
            String line = reader.readLine();
            int currentLine = 0;
            int message = 0;
            while (line != null) {
                //If there are no more bits to encode skip to end
                if (currentLine >= binString.length() ){
                    writer.write(line);
                    writer.newLine();
                    // read next line
                    line = reader.readLine();
                    continue;
                }
                else

                //get bit value from bit string
                 message =
Character.getNumericValue(binString.charAt(currentLine));

                if(message == 0){
                    line = line + " ";

                }
                else if(message == 1){
                    line = line + "  ";
                }
                else{
                    throw new IOException();
                }
                currentLine++;





                // Store amended line in output file
                writer.write(line);
                writer.newLine();
                // read next line
                line = reader.readLine();
            }
            reader.close();
            writer.close();
            System.out.println("hidden");
```

```java
        } catch (IOException e) {
            e.printStackTrace();
        }

    }

    static void retrieve(String inpFile) {
        BufferedReader reader;
        String answer = "";

        try {
            reader = new BufferedReader(new FileReader(inpFile));
            String line = reader.readLine();
            while (line != null) {
                if(line.equals("")){
                    line = reader.readLine();
                    continue;

                }

                else{

                // Your code starts here
                //get last character for each line
                String lastChar = line.substring(line.length()-1);
                //take into account empty line that may have hidden space
                if(lastChar.equals(" ")){
                    if(line.length()==1){
                        answer = answer+"1";
                        line = reader.readLine();
                        continue;
                    }
                    // check if 0 or 1
                    String secondLastChar = line.substring(line.length()-
2);

                    if(secondLastChar.equals("  ")){

                        answer = answer + "1";
                    }
                    else{
                        answer = answer + "0";
                    }
                }





                // read next line
                line = reader.readLine();
            }}

            System.out.println(answer);
            System.out.println("retrieved");
            reader.close();
        } catch (IOException e) {
            e.printStackTrace();
        }

    }
}
```

CODE FOR STEGANO2

```java
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class Stegano2
{
    /**
     * Constructor for objects of class Stegano1
     */
    public Stegano2()
    {
    }

    public static void main(String[] args) {
        String arg1, arg2, arg3, arg4;
        Boolean err = false;

        if (args != null && args.length > 1) { // Check for minimum number
of arguments
            arg1 = args[0];
            arg2 = args[1];

            if (arg2 == "") {
                err = true;
            }
            else if ((arg1.equals("A")) && (args.length > 3)){
                // Get other arguments
                arg3 = args[2];
                arg4 = args[3];
                if (arg3 == "" || arg4 == "") {
                    err = true;
                }
                else {
                    // Hide bitstring
                    hide(arg2, arg3, arg4);
                }
            }
            else if (arg1.equals("E")){
                // Extract bitstring from text
                retrieve(arg2);
            }
            else {
                err = true;
            }
        }
        else {
            err = true;
        }

        if (err == true) {
            System.out.println();
            System.out.println("    Stegano2 <A:E><Input
File><OutputFile><Bitstring>");
            System.out.println("Example: Stegano2 A inp.txt out.txt
0010101");
            System.out.println("Example: Stegano2 E inp.txt");

        }
    }

    static void hide(String inpFile, String outFile, String binString) {
        //
        BufferedReader reader;
        BufferedWriter writer;
        if(binString.length()%2 ==1){
            binString = binString + "0";
        }

        try {
            reader = new BufferedReader(new FileReader(inpFile));
            writer = new BufferedWriter(new FileWriter(outFile));
            String line = reader.readLine();
            int currentLine = 0;
            String message = "";
            int i = 0;
            while (line != null) {
                //get two char from bitstring
                    message = binString.substring(i,i+2);




                //Use spaces ranging from 1 to 4 to indicate binary values
                if(message.equals("00")){
                    line = line + " ";

                }
                else if(message.equals("01")) {
                    line = line + "  ";
                }
                else if(message.equals("10")){
                    line = line + "   ";
                }
                else if(message.equals("11")){
                    line = line + "    ";
                }






                    else{
                    throw new IOException();
                }
                    //increment to get next two digits
                i+=2;
                // Store amended line in output file
                writer.write(line);
                writer.newLine();
                // read next line
                line = reader.readLine();
            }

            reader.close();
            writer.close();
        } catch (IOException e) {
            e.printStackTrace();
        }

    }

    static void retrieve(String inpFile) {
        BufferedReader reader;
        String answer = "";

        try {
            reader = new BufferedReader(new FileReader(inpFile));
            String line = reader.readLine();
            while (line != null) {
                if(line.length()>0){//only runs when there are words
                    //count number of spaces and assign binary values
accordingly
                    if(line.charAt(line.length()-1)== ' '){
                        if(line.charAt(line.length()-2)== ' '){
                            if(line.charAt(line.length()-3)== ' '){
                                if(line.charAt(line.length()-4)== ' '){

                                    answer = answer + "11";
                                }else

                                answer = answer + "10";
                            }
                            else

                            answer = answer + "01";
                        }
                        else
                        answer = answer + "00";
                    }
                }





                // read next line
                line = reader.readLine();
            }

            System.out.println(answer);
            reader.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
```

```
          }
}
```