```java
package tellscopeV4;

import java.awt.BorderLayout;

public class TellServerGui04 extends JFrame {

    /*
     * calcResults array - this array stores the results of the calculations in the following order:
     *
     * index - description
     * 0 - focal length
     * 1 - tube diameter
     * 2 - distance to secondary
     * 3 - secondary size minor
     * 4 - secondary size major
     * 5 - minimum magnitude
     * 6 - minimum resolution
     * 7 - maximum visible magnification
     * 8 - minimum visible magnification
     * 9 - eyepiece magnification
     * 10 - keyword "results" to tell client to sort and display results
     *
     *  */


    /**
     *
     */
    private static final long serialVersionUID = 1L;

    /****** RESULTS STORAGE *************************************/
    protected static String[] calcResults = new String[11];             //array to store processed results

    protected static String resultTitles[] =                            //array to store results titles - test purpose only!!!!
            {   "Focal Length: ",
                "Tube Diamter: ",
                "Distance to Secondary: ",
                "Secondary Size Minor: ",
                "Secondary Size Major: ",
                "Minimum Magnitude: ",
                "Minimum Resolution: ",
                "Maximum Visible Magnification: ",
                "Minimum Visible Magnification: ",
                "Eyepiece Magnification: ",
                "Keyword For Clientside Processing: ",};

    /* class attributes and variables */
    protected static boolean completedResults = false;                  //boolean value used to check if results calculations have been completed
    private static int port = 1234;                                     //default port number value


    private static ServerSocket ss;                                         //server socket to allow client to connect


    /**** LABELS ************************************************************************************************/

    /* input value labels */
    private static JLabel lblLensInput;                                 //label for lens diameter input
    private static JLabel lblFocalInput;                                //label for focal ratio input
    private static JLabel lblEyeInput;                                  //label for eyepiece focal length

    /* result value labels */
    private static JLabel lblFocalLength;                               //label for focal length result
    private static JLabel lblTubeDiameter;                              //label for tube diameter result
    private static JLabel lblDistToSecond;                              //label for distance to secondary result
    private static JLabel lblSecondSizeMaj;                             //label for secondary size major result
    private static JLabel lblSecondSizeMin;                             //label for secondary size minor result
    private static JLabel lblMinMagnitude;                              //label for minimum magnitude result
    private static JLabel lblMinResolution;                             //label for minimum resolution
    private static JLabel lblMaxVisMag;                                 //label for maximum visible magnification
    private static JLabel lblMinVisMag;                                 //label for minimum visible magnification
    private static JLabel lblEyePieceMag;                               //label for eyepiece magnification

    /**** END LABELS ********************************************************************************************/

    /**** TEXTFIELDS + TEXTAREAS ********************************************************************************/

    /* input value text fields */
    private static JTextField txtLensInput;                                 //textField for lens diameter input
    private static JTextField txtFocalInput;                                //textField for focal ratio input
    private static JTextField txtEyeInput;                                  //textField for eyepiece focal length

    /* result value text fields */
    private static JTextField txtFocalLength;                               //textField for focal length result
    private static JTextField txtTubeDiameter;                              //textField for tube diameter result
    private static JTextField txtDistToSecond;                              //textField for distance to secondary result
    private static JTextField txtSecondSizeMaj;                             //textField for secondary size major result
    private static JTextField txtSecondSizeMin;                             //textField for secondary size minor result
    private static JTextField txtMinMagnitude;                              //textField for minimum magnitude result
    private static JTextField txtMinResolution;                             //textField for minimum resolution
    private static JTextField txtMaxVisMag;                                 //textField for maximum visible magnification
    private static JTextField txtMinVisMag;                                 //textField for minimum visible magnification
```

```
101     private static JTextField txtEyePieceMag;                                   //textField for eyepiece magnification
102
103     protected static JTextArea consoleView;                                     //textArea to display console messages
104
105     /**** END TEXTFIELDS + TEXTAREAS ***************************************************************************/
106
107
108     /**** BUTTONS ***********************************************************************************************/
109
110     //private static JButton btnClearButton;                                    //button to clear results set
111
112     /**** END BUTTONS *******************************************************************************************/
113
114
115     /* main method */
116     public static void main(String args[])
117     {
118         new TellServerGui04();                          //create new tell server gui
119     }
120
121     /* default TellServerGui04 constructor */
122     TellServerGui04()
123     {
124
125         /* set up JFrame */
126         this.setSize(500, 400);                                             //set default JFrame size
127         this.setTitle("TellScope Server");                                      //set JFrame title
128         this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);                    //set default close operation
129         this.setLocationRelativeTo(null);                                       //center the frame on screen
130         this.setVisible(true);
131
132         /* create panels */
133
134         //mainPanel
135         JPanel mainPanel = new JPanel();                                        //initialise main panel
136             mainPanel.setLayout(new GridBagLayout());                           //set layout for main panel
137
138         //inputsPanel
139         JPanel inputsPanel = new JPanel();                                      //initialise inputs panel
140             inputsPanel.setLayout(new GridBagLayout());                         //set layout for inputs panel
141
142         //consolePanel
143         JPanel consolePanel = new JPanel();                                     //initialise console panel
144             consolePanel.setLayout(new GridBagLayout());                        //set layouts for console panel
145
146         //resultsPanel
147         JPanel resultsPanel = new JPanel();                                     //initialise results panel
148             resultsPanel.setLayout(new GridBagLayout());                        //set layouts for results panel
149
150         /* create borders for panels */
151
152         Border inputsBorder = BorderFactory.createTitledBorder("Inputs");       //create border for inputs panel
153             inputsPanel.setBorder(inputsBorder);                                //add the border to the inputs panel
154
155         Border consoleBorder = BorderFactory.createTitledBorder("Console");     //create border for console panel
156             consolePanel.setBorder(consoleBorder);                              //add the border to the console panel
157
158         Border resultsBorder = BorderFactory.createTitledBorder("Results");     //create border for results panel
159             resultsPanel.setBorder(resultsBorder);                              //add the border to the results panel
160
161
162         /***** GRIDBAG LAYOUT CONSTRAINTS ***********************************************/
163
164         /* create new grid bag constraints object to help set components in place */
165         GridBagConstraints gc = new GridBagConstraints();   //create new grid constraints object
166         gc.gridx = 0;                                       //set default grid x
167         gc.gridy = 0;                                       //set default grid y
168         gc.gridwidth = 1;                                   //set default grid width
169         gc.gridheight = 1;                                  //set default grid height
170         gc.weightx = 100.0;                                 //set default row width
171         gc.weighty = 100.0;                                 //set default row height
172         gc.insets = new Insets(5,25,15,25);                 //set default padding
173         gc.anchor = GridBagConstraints.WEST;                //set default alignment if component does not fill space
174         gc.fill = GridBagConstraints.NONE;                  //set default fill value (?fill available space)
175
176
177         /************************************************************************/
178
179         /**** INPUT PANEL SECTION ***************************************************/
180
181         /* initialise input labels */
182         lblLensInput = new JLabel("Lens Diameter");                     //initialise JLabel with title"Lens diameter"
183         lblFocalInput = new JLabel("Focal Ratio");                      //initialise JLabel with title"Lens diameter"
184         lblEyeInput = new JLabel("Eyepiece Focal Length");              //initialise JLabel with title"Lens diameter"
185
186         /* initialise input text fields */
187         txtLensInput = new JTextField(10);                              //create new text field, set number of columns
188             txtLensInput.setEditable(false);                            //make textField un-editable
189         txtFocalInput = new JTextField(10);                             //create new text field, set number of columns
190             txtFocalInput.setEditable(false);                           //make textField un-editable
191         txtEyeInput = new JTextField(10);                               //create new text field, set number of columns
192             txtEyeInput.setEditable(false);                             //make textField un-editable
193
194         gc.insets = new Insets(5,10,15,10);
```

```
195         /* add the components to the panel */
196         //add labels
197         inputsPanel.add(lblLensInput, gc);
198         gc.gridy = 1;
199         inputsPanel.add(lblFocalInput, gc);
200         gc.gridy = 2;
201         inputsPanel.add(lblEyeInput, gc);
202 
203         //add text fields
204         gc.gridx = 1;
205         gc.gridy = 0;
206         inputsPanel.add(txtLensInput, gc);
207         gc.gridy = 1;
208         inputsPanel.add(txtFocalInput, gc);
209         gc.gridy = 2;
210         inputsPanel.add(txtEyeInput, gc);
211 
212 
213         /**** END INPUT PANEL SECTION *****************************************************/
214 
215 
216         /**** OUTPUT PANEL SECTION ********************************************************/
217 
218         /* initialise results labels */
219         lblFocalLength = new JLabel("Focal Length");                                    //label for focal length result
220         lblTubeDiameter = new JLabel("Tube Diameter");                                  //label for tube diameter result
221         lblDistToSecond = new JLabel("Distance to Second");                             //label for distance to secondary
    result
222         lblSecondSizeMaj = new JLabel("Second Size Major");                             //label for secondary size major
    result
223         lblSecondSizeMin = new JLabel("Second Size Minor");                             //label for secondary size minor
    result
224         lblMinMagnitude = new JLabel("Min Magnitude");                                  //label for minimum magnitude result
225         lblMinResolution = new JLabel("Min Resolution");                                //label for minimum resolution
226         lblMaxVisMag = new JLabel("Max Visible Magnification");                         //label for maximum visible
    magnification
227         lblMinVisMag = new JLabel("Min Visible Magnification");                         //label for minimum visible
    magnification
228         lblEyePieceMag = new JLabel("Eyepiece Magnification");                          //label for eyepiece magnification
229 
230         /* initialise results text fields */
231         txtFocalLength = new JTextField(10);                                            //text field for focal length result
232             txtFocalLength.setEditable(false);                                          //make textField un-editable
233         txtTubeDiameter = new JTextField(10);                                           //text field tube diameter result
234             txtTubeDiameter.setEditable(false);                                         //make textField un-editable
235         txtDistToSecond = new JTextField(10);                                           //text field distance to secondary
    result
236             txtDistToSecond.setEditable(false);                                         //make textField un-editable
237         txtSecondSizeMaj = new JTextField(10);                                          //text field secondary size major
    result
238             txtSecondSizeMaj.setEditable(false);                                        //make textField un-editable
239         txtSecondSizeMin = new JTextField(10);                                          //text field secondary size minor
    result
240             txtSecondSizeMin.setEditable(false);                                        //make textField un-editable
241         txtMinMagnitude = new JTextField(10);                                           //text field minimum magnitude result
242             txtMinMagnitude.setEditable(false);                                         //make textField un-editable
243         txtMinResolution = new JTextField(10);                                          //text field minimum resolution
244             txtMinResolution.setEditable(false);                                        //make textField un-editable
245         txtMaxVisMag = new JTextField(10);                                              //text field maximum visible
    magnification
246             txtMaxVisMag.setEditable(false);                                            //make textField un-editable
247         txtMinVisMag = new JTextField(10);                                              //text field minimum visible
    magnification
248             txtMinVisMag.setEditable(false);                                            //make textField un-editable
249         txtEyePieceMag = new JTextField(10);                                            //text field eyepiece magnification
250             txtEyePieceMag.setEditable(false);                                          //make textField un-editable
251 
252 
253         /* add components to results panel */
254         //reset layout constraints
255         gc.insets = new Insets(5,10,15,10);
256         gc.gridx = 0;
257         gc.gridy = 0;
258 
259         //add first 5 labels
260         resultsPanel.add(lblFocalLength, gc);
261         gc.gridy = 1;
262         resultsPanel.add(lblTubeDiameter, gc);
263         gc.gridy = 2;
264         resultsPanel.add(lblDistToSecond, gc);
265         gc.gridy = 3;
266         resultsPanel.add(lblSecondSizeMin, gc);
267         gc.gridy = 4;
268         resultsPanel.add(lblSecondSizeMaj, gc);
269 
270         //add first five text fields
271         gc.gridx = 1;
272         gc.gridy = 0;
273         resultsPanel.add(txtFocalLength, gc);
274         gc.gridy = 1;
275         resultsPanel.add(txtTubeDiameter, gc);
276         gc.gridy = 2;
277         resultsPanel.add(txtDistToSecond, gc);
278         gc.gridy = 3;
```

```
279         resultsPanel.add(txtSecondSizeMin, gc);
280         gc.gridy = 4;
281         resultsPanel.add(txtSecondSizeMaj, gc);
282 
283         //add final 5 labels
284         gc.gridx = 2;
285         gc.gridy = 0;
286         resultsPanel.add(lblMinMagnitude, gc);
287         gc.gridy = 1;
288         resultsPanel.add(lblMinResolution, gc);
289         gc.gridy = 2;
290         resultsPanel.add(lblMaxVisMag, gc);
291         gc.gridy = 3;
292         resultsPanel.add(lblMinVisMag, gc);
293         gc.gridy = 4;
294         resultsPanel.add(lblEyePieceMag, gc);
295 
296         //add final 5 text fields
297         gc.gridx = 3;
298         gc.gridy = 0;
299         resultsPanel.add(txtMinMagnitude, gc);
300         gc.gridy = 1;
301         resultsPanel.add(txtMinResolution, gc);
302         gc.gridy = 2;
303         resultsPanel.add(txtMaxVisMag, gc);
304         gc.gridy = 3;
305         resultsPanel.add(txtMinVisMag, gc);
306         gc.gridy = 4;
307         resultsPanel.add(txtEyePieceMag, gc);
308 
309         /**** END OUTPUT PANEL SECTION ************************************************/
310 
311         /**** CONSOLE PANEL SECTION ***************************************************/
312 
313         consolePanel.setLayout(new BorderLayout());
314         consoleView = new JTextArea();
315             consoleView.setRows(7);
316             consoleView.setColumns(22);
317             consoleView.setEditable(false);
318 
319         JScrollPane scroll = new JScrollPane(consoleView);
320         consolePanel.add(scroll, BorderLayout.CENTER);
321 
322 
323         /**** END CONSOLE PANEL SECTION ***********************************************/
324 
325 
326         /**** ADD SUB PANELS TO MAIN PANEL ********************************************/
327         gc.insets = new Insets(5,10,10,10);
328         gc.gridx = 0;
329         gc.gridy = 0;
330 
331         mainPanel.add(inputsPanel, gc);
332 
333         gc.gridx = 0;
334         gc.gridy = 1;
335         gc.gridwidth = 2;
336 
337         mainPanel.add(resultsPanel, gc);
338 
339         //add console window panel
340         gc.gridx = 1;
341         gc.gridy = 0;
342         gc.gridwidth = 1;
343         mainPanel.add(consolePanel, gc);
344 
345         this.add(mainPanel);
346         this.pack();
347         this.setResizable(false);
348         this.setVisible(true);
349 
350         /**** END ADD SUB PANELS TO MAIN PANEL ****************************************/
351 
352     /* call startServer() method to start the server running */
353     startServer();
354 
355 
356     }//end TellServer04 constructor
357 
358 
359     /***** PUBLIC METHODS *************************************************************/
360 
361     //test method to print results array
362     public static void printResults()
363     {
364         for(int i=0;i<11;i++)
365         {
366             System.out.println(resultTitles[i] + calcResults[i]);
367         }
368 
369     }
370 
371     /****************************/
372 
```

```
373     /***** END PUBLIC METHODS ********************************************************************/
374
375
376
377     /***** PROTECTED METHODS **********************************************************************/
378
379     //sendResultsToClient method - this method starts a thread to send the results to the client
380
381         protected static void sendResultsToClient()
382         {
383             //create thread
384
385             completedResults = true;
386
387         }
388
389         /***************************/
390
391         protected static void dontSendResultsToClient()
392         {
393             //create thread
394
395             completedResults = false;
396
397         }
398
399         /***************************/
400
401         protected static void setInputs(String lensDiam, String focalRatio, String eyeMag)
402         {
403
404             txtLensInput.setText(lensDiam);
405
406             txtFocalInput.setText(focalRatio);
407
408             txtEyeInput.setText(eyeMag);
409
410         }
411
412
413         protected static void setResults()
414         {
415             /* set results values */
416             txtFocalLength.setText(calcResults[0]);          //set focal length result
417             txtTubeDiameter.setText(calcResults[1]);         //set tube diam result
418             txtDistToSecond.setText(calcResults[2]);         //set dist to second result
419             txtSecondSizeMin.setText(calcResults[3]);        //set second size min result
420             txtSecondSizeMaj.setText(calcResults[4]);        //set second size maj result
421             txtMinMagnitude.setText(calcResults[5]);         //set min magnitude result
422             txtMinResolution.setText(calcResults[6]);        //set min res result
423             txtMaxVisMag.setText(calcResults[7]);            //set max vis mag result
424             txtMinVisMag.setText(calcResults[8]);            //set min vis mag result
425             txtEyePieceMag.setText(calcResults[9]);          //set eyepiece mag result
426         }
427
428
429
430         /***** END PROTECTED METHODS ******************************************************************/
431
432
433         /**** PRIVATE METHODS *************************************************************************/
434
435
436         private static void startServer()
437         {
438             /* start server and wait for connection to be made */
439             try                                                                    //try catch block to catch
    network connection errors
440             {
441                 System.out.println("Tells Server");                             //print message to console
442                 System.out.println("Listening on port: " + port);               //print port number to console
443                 consoleView.append("TellScope Server");
444                 consoleView.append("\nListening on port: " + port);
445                 ss = new ServerSocket(port);                                   //initialise server socket
446
447                 while(true)                                                    //while loop to keep server
    running
448                 {
449                     Socket s = ss.accept();                                    //create socket to client when
    new client connects
450                     System.out.println("Connection established!");             //print message to console when
    new client connects
451
452                     /* create a new thread to wait for input, perform calculations and start sendResults thread */
453
454                     Thread calculateThread = new Thread(new TellCalcThread(s));         //instantiate new telescope
    calculation thread
455                     calculateThread.start();                                   //start new telescope calculation
    thread
456
457                 }//end while loop
458
459             }
460             catch(Exception e)                                                 //catch any exceptions
```

```
461             {
462                 System.out.println("System exception!");                                 //print error message
463 
464             }
465 
466         }//end startSever() method
467 
468 
469         /**** END PRIVATE METHODS ***********************************************************************/
470 
471 }//end TellServer04 class
472 
```