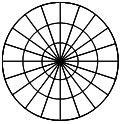
|  |  |
| --- | --- |
|  | |
| Create Dartboard and Design Applets | |
| **Updated** | **10.28.2018 2:55 PM** |

# **Problem 1:**

Develop (including designing, coding and testing) an applet that displays a dartboard pattern like the one shown here, with equally spaced concentric circles and radial lines. The applet must check the available width and height, so that the pattern is drawn as large as possible. There should always be three equally spaced circles, but the number of lines must be specified by a parameter in the HTML file. The number of lines should be between 2 and 30, and should default to 20 if outside of these limits.

**Note:** *a point at angle x on the circumference of a unit circle has coordinates cos(x), sin(x).*



# **Approach**

The HTML page will include a parameter to set the Height, Width and number of radial lines to be used. The Applet will check the size of the window and using the center point and radius draw the largest circle possible. Within this two further equidistant circles are the draw. The Applet will check the parameter for the number of radial lines does not exceed 30 or below 2. If the parameter fails this test then the amount of lines if set to 20. The applet then loops the desired number of lines using the center for the first set of coordinates and using angle x on the circle with coordinates cos(x), sin(y).

# **Design**

Set default size for the window

Set default number of radial lines

Initialization takes the number of radial lines from the HTML parameter

Set Size and Set to Visible

Construct a new Graphics2D object.

Check the window width and find center point for X

Check the window height and find center point for Y

Set center point value pair (x,y)

Set circle radius based on center point so that the pattern is drawn as large as possible.

Calculate the three concentric concentric circle

Validate if the number of radials is under 2 or over 30 and set to 20 if it is

Loop number of times set by parameter or default to create radial lines using angle x on the circle with coordinates cos(x), sin(y).

EXIT

# **Testing**

The program was run several times with different inputs to ensure the code behaves as expected for each possible execution scenario including the input of invalid and out of bound parameters. The program was also tested for scaling.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Test** | **Expected Result** | **Actual Result** |
| 1 | Set radial line parameter to a number between 2 and 20. | A dartboard pattern with that amount of lines is drawn as large as possible | A dartboard pattern with that amount of lines is drawn as large as possible |
| 2 | Set radial line parameter to a number below 2 or above 20. | A dartboard pattern with that 20 lines is drawn as large as possible | A dartboard pattern with that 20 lines is drawn as large as possible |
| 3 | Do not set the radial line parameter. | A dartboard pattern with that 20 lines is drawn as large as possible | A dartboard pattern with that 20 lines is drawn as large as possible |
| 4 | Increase and decrease the window size and check the dartboard pattern responds. | The dartboard pattern responds by resizing | The dartboard pattern responds by resizing |

# **Source Code – DartsBoard.java**

|  |
| --- |
| /\* mmcmahon\_wk3\_assignments\_Question1  \* Develop (including designing, coding and testing) an applet that displays a dartboard pattern like the one shown here,  \* with equally spaced concentric circles and radial lines. The applet must check the available width and height, so that the  \* pattern is drawn as large as possible. There should always be three equally spaced circles, but the number of lines must be  \* specified by a parameter in the HTML file. The number of lines should be between 2 and 30, and should default to 20 if outside  \* of these limits.  \* References  \* Draw a circle with a radius and points around the edge. (2016). Stackoverflow.com. Retrieved 11 September 2016, from  \* http://stackoverflow.com/questions/2508704  \* JavaFX - JavaFX - Dartboard with Shapes. (2013). Hameister.org. Retrieved 11 September 2016, from  \* http://www.hameister.org/JavaFX\_Dartboard.html  \*/  import java.applet.Applet; //import Applet package  import java.awt.\*; //import Abstract Windowing Toolkit (AWT) package  public class DartsBoard extends Applet {  public int center\_x, center\_y; //Variables for the center x, y coordinates  public int no\_Segment = 20; //Set default number of radial lines  public int r, r2, r3; //Radius values    /\* initialization takes the no\_Segment as the number of radial lines from the HTML parameter \*/  public void init (int no\_Segment) {  setVisible(true); //Set to visible  }  /\*Applet paints itself in the browser\*/  public void paint(Graphics g) {  Graphics2D g2d = (Graphics2D) g; // Construct a new Graphics2D object.    int center\_x = getWidth() / 2; //Get Width and find center point for X  int center\_y = getHeight() / 2; //Get Height and find center point for Y  int middle = Math.min(center\_x, center\_y); //Set center point value pair (x,y)  int r = 4 \* middle / 4; //Set circle radius based on center point so that the pattern is drawn as large as possible.  int r2 = r / 2; //Set second concentric circle  int r3 = r / 5; //Set third concentric circle  g2d.setColor(Color.black); //Set the line colours to black    g2d.drawOval(center\_x - r, center\_y - r, 2 \* r, 2 \* r); //Draw the main circle  g2d.drawOval(center\_x - r2, center\_y - r2, 2 \* r2, 2 \* r2); //Draw the second circle  g2d.drawOval(center\_x - r3, center\_y - r3, 2 \* r3, 2 \* r3); //Draw the third circle    //Validate if the number of radials is under 2 or over 30 and set to 20 if it is  if(no\_Segment < 2 || no\_Segment > 30){  this.no\_Segment = 20;  }    //Loop no\_segment times to create radial lines using angle x on the circle with coordinates cos(x), sin(y).  for (int i = 0; i < no\_Segment; i++) {  double t = 2 \* Math.PI \* i / no\_Segment;  int x = (int) Math.round(center\_x + r \* Math.cos(t));  int y = (int) Math.round(center\_y + r \* Math.sin(t));  g2d.drawLine(center\_x, center\_y, x, y);  }  }  } |

# **Source Code – DartsBoard.html**

|  |
| --- |
| <html>  <body>  <h1>Dart Board Applet!</h1>  <p>  <applet code=DartsBoard.class width=500 height=500>  <param name=no\_Segment value="20">  </applet>  </body>  </html> |

# **Problem 2:**

Write a Java applet that just displays a static picture or design (e.g. a smiley face, a sunburst pattern) made from ovals, lines, rectangles, and/or graphics commands you have found in your research. Consider using loops, etc., for repeated patterns. Submit up a screen-grab of your applet along with the code. Be original!

# **Approach**

I used ovals, lines, rectangles, and graphics commands to create a line drawing similar to that which appeared in the TV series Breaking Bad of Heisenberg.

# **Design**

Set default window size

Set to visible

Create text above the image to say HEISENBERG

Set the fill colour to black

Create a rectangle for the Hat

Create a line for the Hat brim

Using an oval draw the Face

Add two Forehead Lines

Use an Oval to create the Left Eye glass

Use a line to create Nose Bridge of glasses

Use an Oval to create the Right Eye glass

Using three lines draw triangle for the nose on the face

Use a loop of line to create a moustache

Use a line for the Mouth

Write a text line under the face "I AM THE ONE WHO KNOCKS!"

Get returns for all variables

EXIT

# **Testing**

Since this was a small code assignment it did not require a lot of testing

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Test** | **Expected Result** | **Actual Result** |
| 1 | The code will draw a face and add text when run | The code will draw a face and add text when run | The code will draw a face and add text when run |

# **Source Code – BreakingBad.java**

|  |
| --- |
| /\* mmcmahon\_wk3\_assignments\_Question2  \* Write a Java applet that just displays a static picture or design (e.g. a smiley face, a sunburst pattern)  \* made from ovals, lines, rectangles, and/or graphics commands you have found in your research. Consider using  \* loops, etc., for repeated patterns. Submit up a screen-grab of your applet along with the code. Be original!  \*  \*/  **import** java.applet.\*;  **import** java.awt.\*;  **public** **class** BreakingBad **extends** Applet{    **public** **void** init(){  setSize(300, 300); //set default size  setVisible(**true**); //set to visable  }    **public** **void** paint(Graphics g){    g.drawString("HEISENBERG", 107, 75); //Create text above the image    g.setColor(Color.***black***); // Set the fill colour  g.fillRect(112, 85, 75, 25); //create a rectangle for the Hat  g.drawLine(90,110,210,110); //create a line for thd Hat brim    g.drawOval(105,90,90,115); //draw the Face    g.drawLine(135,115,165,115); //add a Forehead Line 1  g.drawLine(130,120,170,120); //add a Forehead Line 2    g.fillOval(115,125,30,20); //create the Left Eye glass  g.drawLine(135,135,165,135); // nose bridge of glasses  g.fillOval(155,125,30,20); //crate the Right Eye glass    // Draw a Nose on the face  g.drawLine(145,165,150,140);  g.drawLine(155,165,150,140);  g.drawLine(145,165,155,165);    //loop through a line to create a moustache  **int** mwidth = 132;  **for**(**int** i = 0; i < 34; i++){  mwidth = mwidth + 2;  g.drawLine(mwidth,170,mwidth,178); //moustache  i++;  }    //add a line for the Mouth  g.drawLine(135,182,165,182); //Mouth    //Write a text line under the face  g.drawString("I AM THE ONE WHO KNOCKS!", 70, 225);    }    } |

# **Applet Output**



# **References:**

Draw a circle with a radius and points around the edge. (2016). Stackoverflow.com. Retrieved 11 September 2016, from http://stackoverflow.com/questions/2508704

JavaFX - JavaFX - Dartboard with Shapes. (2013). Hameister.org. Retrieved 11 September 2016, from http://www.hameister.org/JavaFX\_Dartboard.html