

**Basketball Game**

**Mini-Project 4: Angle Gauge**

In this fourth mini-project, you'll code the angle gauge. The gauge won't change based on arrow keys yet. This is just the drawing of the gauge. You'll import this code in two different places in activity 5.

**Objectives:**

- create a variable  $\theta$  to store the angle
- use the Line function to create an angle meter

**Basketball Game Project Overview:**

After completing a series of 8 mini projects, you will have a basketball game similar to the one on the right. The code for projects 1 -4 will be imported into project 5. Projects 6-8 will build upon project 5.

Mini-Project Order:

1. Draw the Background
2. Draw the Net
3. Power Gauge
- 4. Angle Gauge**
5. Merge the Projects and Code the Arrow Keys
6. Toss the Ball
7. The Game
8. Win the Trophy

After Project 6



Angle Gauge



**Teacher Tip:**

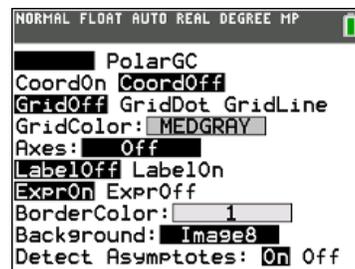
This step will be imported twice in the basketball game. This picture needs to be redrawn each time the student changes the angle, therefore, it is important to code and error trap this separately before it is imported into the final project.

This project uses some right triangle trigonometry to rotate a line. If students haven't learned this material yet, that is ok it will be taught and all formulas will be provided.

1. It will be useful to have your background loaded as reference. We won't do this in code because it is already coded in our NET code you use in later projects.

Press 2<sup>nd</sup> Zoom (   )

In Background, select the image that contains your basketball background.

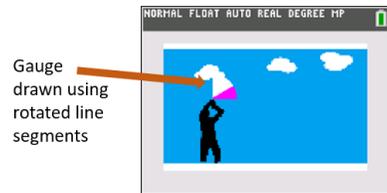


2. Create a new program named ARC.

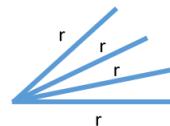
Add a comment line and a ClrDraw line. It will be beneficial to have the comment line when compiling the final project.

```
:"ANGLE GAUGE
:ClrDraw
```

You will create the angle gauge by drawing a series of line segments similar to the ones on the power gauge. Instead of parallel lines like the power gauge, these lines will need to radiate from a single point as they are drawn.

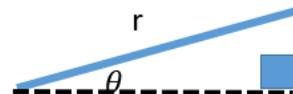


3. Each line segment of the arc has the same radius



You will use some right triangle trigonometry to find the horizontal and vertical displacement of each line segment.

Do you know how to find the height and width of the triangle on the right?



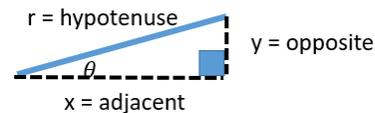
**Teacher Tip:**

If students haven't explored right triangle trigonometry, that is ok. The math and the formula to use will be given in the next steps.

4. In a right triangle:

$$\sin(\theta) = \frac{\textit{opposite}}{\textit{hypotenuse}}$$

$$\cos(\theta) = \frac{\textit{adjacent}}{\textit{hypotenuse}}$$



Therefore,

$$\sin(\theta) = \frac{y}{r}$$

$$\cos(\theta) = \frac{x}{r}$$

Solve the first equation for y.

Solve the second equation for x.

5.  $r \cdot \sin(\theta) = y$        $r \cdot \cos(\theta) = x$

We will use this relationship to draw the line segments in the angle gauge.

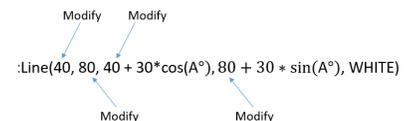
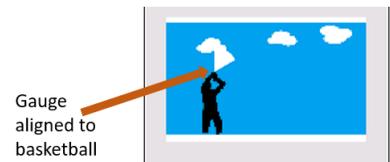
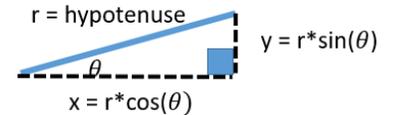
```
:Line(X1, Y1, X1 + r*cos(θ), Y1 + r*sin(θ), color)
```

You will start with a radius of 30. You may modify this value to fit your picture.

```
:Line(X1, Y1, X1 + 30*cos(θ), Y1 + 30*sin(θ), color)
```

6. Add the following code to your program. Execute your program. If the gauge is not aligned to the basketball, modify the x and y values. Make sure to modify *both* x's and *both* y's in the Line( statement (40's and 80's in the example).

```
:45 → θ
:For(A, 90, θ, -1)
:Line(40, 80, 40 + 30*cos(A°), 80 + 30*sin(A°), WHITE)
:End
```



**Teacher Tip:**

By inserting the degree symbol in the formula, it will ensure the calculations are done in degree mode even if the calculator is in radian mode.

7. Now to complete the angle gauge. You will indicate the angle  $\theta$  using different color. Add the code in bold to your project.

```
: 45 → θ
:For(A, 90, θ, -1)
:Line(65, 95, 65 + 30*cos(A°), 95 + 30 * sin(A° ), WHITE)
:End
:For(A, θ, 0, -1)
:Line(65, 95, 65 + 30*cos(A° ), 95 + 30*sin(A° ), MAGENTA)
:End
```



Be careful, make sure all the thetas are 'θ' and all the zeros are 0.

**\*If you choose to skip the next three slides, delete the ClrDraw from the top of the program after you have thoroughly tested your code.**



### Teacher Tip:

Step 9 and 10 are purely optional based on student preference. Students will find it easier to make a shot if they have the different color reference points while playing.

#### 8. Option 1 Step 9

Entire meter changes color based on the value of  $\theta$ .



#### Option 2 Step 10

The meter has different color segments based on the value of  $\theta$ .



9. Option 1: The entire angle gauge color is based on the value of  $\theta$ . Therefore, you'll use multiple Ifs to select the color before the For statement. Based on the results of the If, you'll set the color. You get to decide how many Ifs to include which will determine the number of color choices. Your modification should look something like the lines below

```

:MAGENTA →C
:If  $\theta > 33$ 
:Then
:NAVY →C
:End
:If  $\theta > 66$ 
:Then
:GREEN →C
:End
:For(A, 90,  $\theta$ , -1)
:Line(40, 80, 40 + 30*cos(A ° ), 80 + 30 * sin(A ° ), WHITE)
:End
:For(A,  $\theta$ , 0, -1)
:Line(40, 80, 40 + 30*cos(A ° ), 80 + 30 * sin(A ° ), C)
:End

```



**\*Once your program is complete and functioning, delete the ClrDraw from the top of your code.**



# 10 MOC: Beyond Basics

## TI-84 PLUS CE TECHNOLOGY

## BASKETBALL GAME: MINI-PROJECT 4

### TEACHER NOTES

10. Option 2: The each layer of the angle gauge is based on a selection. Therefore, the If is inside the For statement. Based on the results of the If, you'll set the color for that line. You get to decide how many Ifs to include which will determine the number of color choices. Your modification should look something like the lines below

```

:For(A, 90,  $\theta$ , -1)
:Line(40, 80, 40 + 30*cos(A°), 80 + 30 * sin(A°), WHITE)
:End
:For(A,  $\theta$ , 0, -1)
:MAGENTA →C
:If A > 33
:Then
:NAVY →C
:End
:If A > 66
:Then
:GREEN →C
:End
:Line(40, 80, 40 + 30*cos(A°), 80 + 30*sin(A°)C)
:End

```

**\*Once your program is complete and functioning, delete the ClrDraw from the top of your code.**



#### Teacher Tip:

Sample Code Slide 5: All Magenta

```

NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha][phat] [f5]
PROGRAM: ARC
:"ARC
:WHITE→C
:30→ $\theta$ 
:For(A, 90,  $\theta$ , -1)
:Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
:End
:MAGENTA→C
:For(A,  $\theta$ , 0, -1)
:Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
:End

```



Sample Code Slide 6: Change Color as Angle Changes



```

NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [a] [pha] [f5]
PROGRAM: ARC9
: "ARC
: WHITE→C
: 30→θ
: For(A, 90, θ, -1)
: Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
: End
: MAGENTA→C
: If θ>33
: Then
: NAVY→C
: End
: If θ>66
: Then
: GREEN→C
: End
: For(A, θ, θ, -1)
: Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
: End

```



Sample Code Slide 7 Different Segments Different Color

```

NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [a] [pha] [f5]
PROGRAM: ARC10
: "ARC
: WHITE→C
: 70→θ
: For(A, 90, θ, -1)
: Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
: End
: For(A, θ, θ, -1)
: MAGENTA→C
: If A>33
: Then
: NAVY→C
: End
: If A>66
: Then
: GREEN→C
: End
: Line(65, 95, 65+30*cos(A°),
95+30*sin(A°), C)
: End

```

