Basketball Game | **Mini-Project 8: The Trophy**

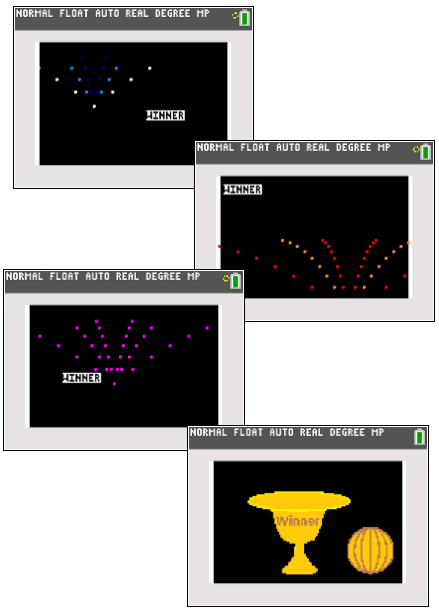
In this final mini project, you'll create fireworks and a trophy to appear after the user sinks enough baskets.

Objectives:

- Create a temporary file WIN
- Use an IF to determine when the player wins
- Use a For loop to repeat code
- Import backgrounds using TI-Connect

The 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 (and fireworks!)**

Teacher Tip: Once the basketball game is complete and the student is sure there aren't any errors, they can delete all the temporary files created. Make sure students don't delete the main game. The demo has three different firework bursts. Students may choose to put as many or as few fireworks as they like.

1. You could add this code directly to the end of the basketball game. However, you'd have to run through the entire game each time you needed to fix a bug or wanted to see your fireworks progress. Therefore, we are going to create a short program that will allow you to design your fireworks and trophy code without all the other lines you have done so far.

   Create a new program named WIN.
   You will import your BACKGRND code from project 1. We import this code so we'll have all the settings set to the same as the game's settings.

   Use recall to import the keyboard code
   rcl (2nd sto→)
   prgm
   EXEC
   Select BACKGRND
   Press the enter key

2. Connect your calculator to your computer and open up TI-Connect.

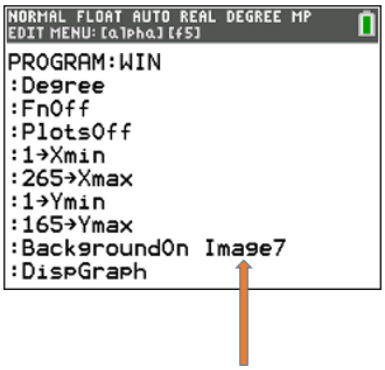
Open the calculator explorer workspace.

Import the night sky as IMAGE7

Import the trophy as IMAGE9

3. Change the IMAGE to Image7 (the night sky).

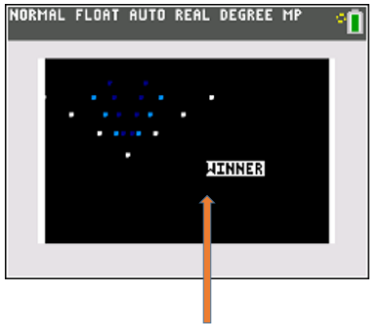
   The image options are located in
   - Vars
   - Picture & Background
   - Background



4. You can place the WINNER text anywhere on the screen. The generic code is: Text(row, column, “text”)

   The text seen at the right starts at row 90 and column 150.

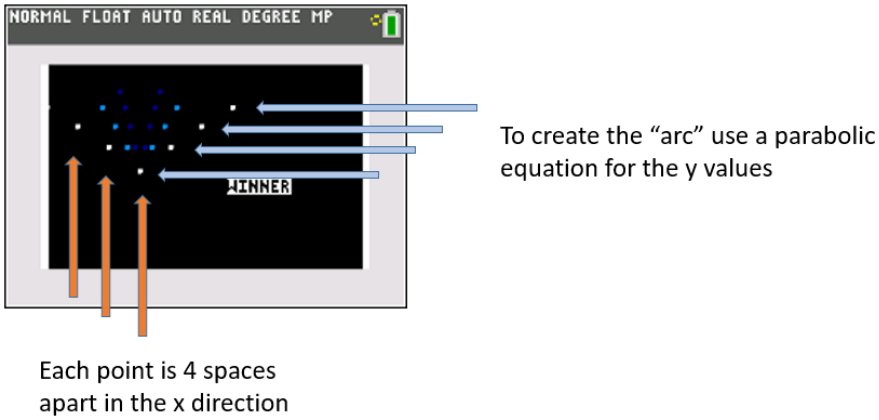
   To draw the WINNER text, you’ll clear the drawing screen then add the text.
   :ClrDraw
   :Text(90,150,“WINNER”)



Start row 90
column 150

Teacher Tip:
Students may choose to add a stretch to the parabolic equation or use it as presented.

5. To draw the fireworks, you'll use the Circle command. Remember, it uses Circle(x, y, radius, color)



6.

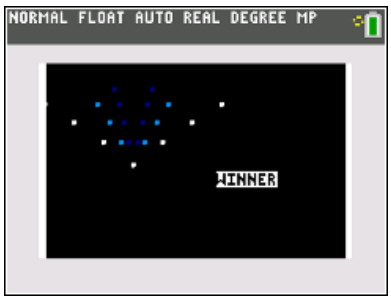
The dark blue equation is Circle(80 + I*4, 180 – (I – 10)^2, 1, NAVY)

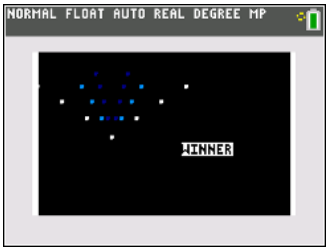
Initial X value

Change X by 4 each time through a loop

Initial Y value

Change Y parabolically each time through a loop



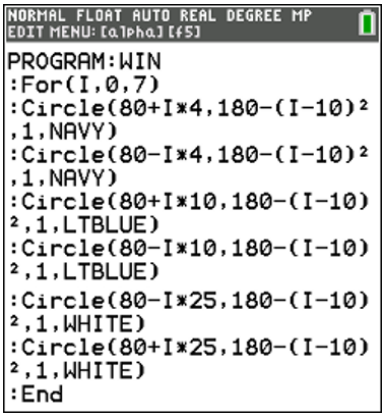


7. The loop for the first firework is

```
:For(I, 0, 7)
:Circle(80 + I*4, 180 – (I-10)², 1, NAVY)
:Circle(80 - I*4, 180 – (I-10)², 1, NAVY)

:Circle(80 + I*10, 180 – (I-10)², 1, LTBLUE)
:Circle(80 - I*10, 180 – (I-10)², 1, LTBLUE)

:Circle(80 + I*25, 180 – (I-10)², 1, WHITE)
:Circle(80 - I*25, 180 – (I-10)², 1, WHITE)
:END
```

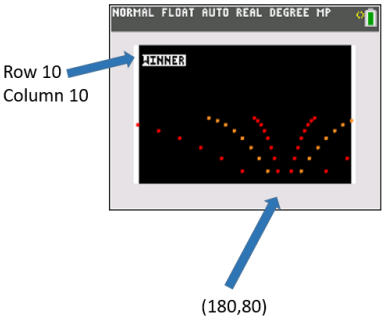


Notice, to make the colors symmetric, one equation adds to 80 the other equation subtracts the same amount from 80.

Execute your code. Debug any errors.

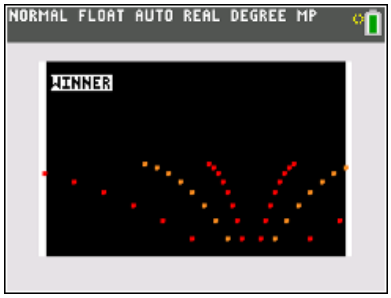
8. To draw the second firework
   - Clear the drawing
   - Draw the WINNER text at (10,10)
   - Center the new fireworks at (180,80)

Can you draw the second firework before looking at the code on the next step?



9. The loop for the second firework is

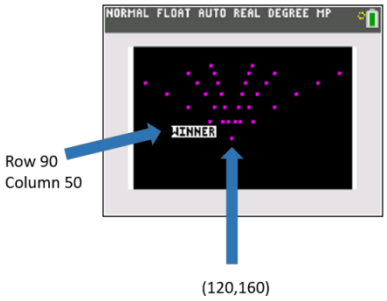
```
:ClrDraw
:For(I, 0, 7)
:Circle(180 + I*4, 80 – (I-10)², 1, RED)
:Circle(180 - I*4, 80 – (I-10)², 1, RED)
:Circle(180 + I*10, 80 – (I-10)², 1, ORANGE)
:Circle(180 - I*10, 80 – (I-10)², 1, ORANGE)
:Circle(180 + I*25, 80 – (I-10)², 1, RED)
:Circle(180 - I*25, 80 – (I-10)², 1, RED)
:END
```



Execute your code. Debug for errors.

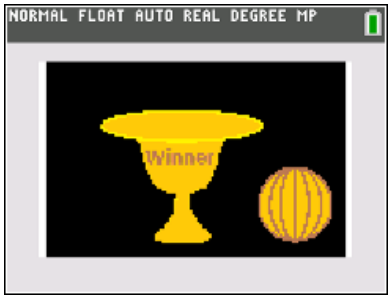
```
NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha] [f5]
PROGRAM:WIN
:
:ClrDraw
:Text(10,10,"WINNER")
:For(I,0,10)
:Circle(180+I*4,80-(I-10)²
,1,RED)
:Circle(180-I*4,80-(I-10)²
,1,RED)
:Circle(180+I*10,80-(I-10)
²,1,ORANGE)
:Circle(180-I*10,80-(I-10)
²,1,ORANGE)
:Circle(180-I*25,80-(I-10)
²,1,RED)
:Circle(180+I*25,80-(I-10)
²,1,RED)
:End
:
```

10. To draw the third firework
    - Clear the drawing
    - Draw the WINNER text at row 90, column 50
    - Center the new fireworks at (120,160)



Teacher Tip: Code for the third firework

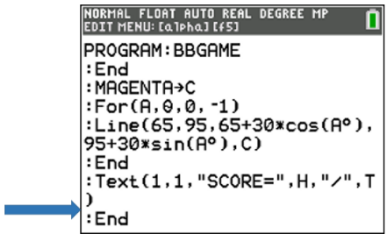
```
NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha] [f5]
PROGRAM:BBGAME
:ClrDraw
:Text(90,50,"WINNER")
:For(I,0,10)
:Circle(120+I*4,160-(I-10)
²,1,MAGENTA)
:Circle(120-I*4,160-(I-10)
²,1,MAGENTA)
:Circle(120-I*10,160-(I-10
)²,1,MAGENTA)
:Circle(120+I*10,160-(I-10
)²,1,MAGENTA)
:Circle(120-I*25,160-(I-10
)²,1,MAGENTA)
:Circle(120+I*25,160-(I-10
)²,1,MAGENTA)
:End
```



11. Last step before you import your code to the game, clear the drawing and display the trophy

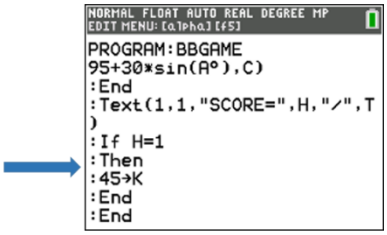
    :ClrDraw
    :BackgroundOn Image9

    Execute your program and debug any errors.



12. Open the code for BBGAME.

    Go to the end of your code.
    Insert 4 blank lines above the last End statement.



13. What constitutes a winning game?
    One basket? Five baskets?
    Is 5 / 6 a winner but 5/100 not?

    For now, you'll say one basket is a winner so we can test the code. Insert the following lines of code. They will set the key value K to 45 which exits the overarching WHILE statement

    :IF H = 1
    :Then
    :45 →K
    :End

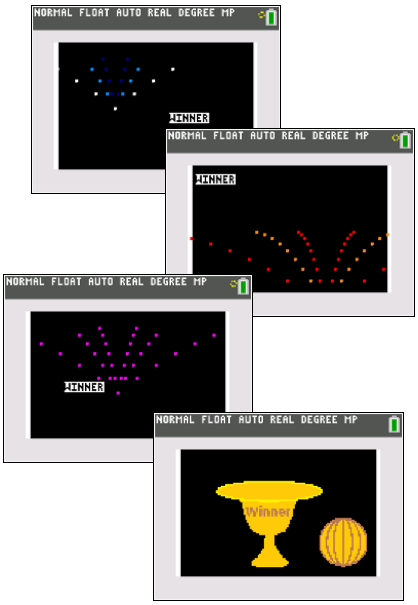
14. Scroll to the end of your code. Fireworks are the last thing that should happen.

    You will import your WIN code

    Use recall to import the keyboard code

    rcl (2nd sto→)
    prgm
    EXEC
    Select WIN
    Press the enter key

15. Play your game a few times.



Change the winning setting to a setting of your choice.

You might say 3 SWISHES are a win, so your IF statement would be IF H = 3.

You might say at least 3 SWISHES and at least 50% are a win, so your IF statement would be:

H/T →P

If H ≥ 3 *and* P $\geq$ 0.5

Teacher Tip:

```
NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha] [f5]
PROGRAM:BBGAME
:0→H
:0→T
:Degree
:FnOff
:PlotsOff
:1→Xmin
:265→Xmax
:1→Ymin
:165→Ymax
:BackgroundOn Image8
:DispGraph
:0→H
:0→T
:Degree
:FnOff
:PlotsOff
:1→Xmin
:265→Xmax
:1→Ymin
:165→Ymax
:BackgroundOn Image8
:DispGraph
:
:ClrDraw
:randInt(30,120)→Y
:Y-10→Z
:Y-5→S
```

```
:Line(245,Y,255,Y,YELLOW)
:Line(245,Z,255,Z,YELLOW)
:Line(240,S,245,Y,YELLOW)
:Line(240,S,245,Z,YELLOW)
:Line(255,Y,260,S,YELLOW)
:Line(255,Z,260,S,YELLOW)
:Y-25→N
:Line(240,Z,250,N,DARKGRAY
)
:Line(245,Z,240,N,DARKGRAY
)
:Line(245,Z,245,N,DARKGRAY
)
:Line(260,Z,260,N,DARKGRAY
)
:Line(260,S,255,N,DARKGRAY
)
:Line(255,Z,250,N,DARKGRAY
)
:Line(250,Z,245,N,DARKGRAY
)
:Line(250,Z,255,N,DARKGRAY
)
:
:"GUAGE
:20→F
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:For(C,1,F)
:Line(12,C+12,19,C+12,MAGE
NTA)
:End
:For(C,F+1,100)
:Line(12,C+12,19,C+12,WHIT
E)
:End
:
:
:"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
:For(A,θ,θ,-1)
:Line(65,95,65+30*cos(A°),
95+30*sin(A°),C)
:End
:
:"ARROW KEYS
```

```
:"GAUGE
:ClrDraw
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:90→F
:MAGENTA→C
:If F>33
:Then
:ORANGE→C
:End
:If F>66
:Then
:BROWN→C
:End
:For(A,1,F)
:Line(12,A+12,19,A+12,C)
:End
:For(A,F+1,99)
:Line(12,A+12,19,A+12,WHIT
E)
:End
:"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
```

```
:"GAUGE
:ClrDraw
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:90→F
:For(A,1,F)
:MAGENTA→C
:If A>33
:Then
:ORANGE→C
:"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
:End
```

```
:"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
```

```
:0→K
:While K≠45
:While K≠105
:getKey→K
:If K=45
:Then
:Stop
:End
:If K=25 and F<98
:Then
:F+2→F
:End
:If K=34 and F>2
:Then
:F-2→F
:End
:If K=26 and θ>2
:Then
:θ-2→θ
:End
:If K=24 and θ<88
:Then
:θ+2→θ
:End
:"REFRESH
:If K=24 or K=26
:Then
:"ARC
:WHITE→C
:For(A,90,θ,-1)
:Line(65,95,65+30*cos(A°),
95+30*sin(A°),C)
:End
:MAGENTA→C
:For(A,θ,0,-1)
:Line(65,95,65+30*cos(A°),
95+30*sin(A°),C)
:End
:End
:If K=25 or K=34
:Then
:"GUAGE
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:For(C,1,F)
:Line(12,C+12,19,C+12,MAGE
NTA)
```

```
:"GAUGE
:ClrDraw
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:90→F
:MAGENTA→C
:If F>33
:Then
:ORANGE→C
:End
:If F>66
:Then
:BROWN→C
:End
:For(A,1,F)
:Line(12,A+12,19,A+12,C)
:End
:For(A,F+1,99)
:Line(12,A+12,19,A+12,WHIT
E)
:End
:"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,0,-1)
:Line(65,95,65+30*cos(A°),
95+30*sin(A°),C)
:End
```

```
:"GAUGE
:ClrDraw
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:90→F
:For(A,1,F)
:MAGENTA→C
:If A>33
:Then
:ORANGE→C
:End
:"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,0,-1)
:MAGENTA→C
:End
```

```
:"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,0,-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
```

```
:End
:End
:
:"SHOOT BASKET
:If K=105
:Then
:Y→G
:DrawF ((95+tan(0°)*(X-65)
-4.9*((X-65)/(F*cos(0°)))²
)/(X>65))
:G→Y
:0→A
:For(X,241,259)
:95+tan(0°)*(X-65)-4.9*((X
-65)/(F*cos(0°)))^2→W
:If W<Y and W>Y-10
:Then
:1→A
:End
:End
:A+H→H
:1+T→T
:TextColor(WHITE)
:If A=1
:Then
:Text(80,80,"SWISH")
:Else
:Text(80,80,"MISS")
:End
:End
:End
:While K≠95
:getKey→K
:If K=95
:Then
:ClrHome
:End
:End
:Degree
:FnOff
:PlotsOff
:1→Xmin
:265→Xmax
:1→Ymin
:165→Ymax
:BackgroundOn Image8
:DispGraph
:
:ClrDraw
:randInt(30,120)→Y
:Y-10→Z
:Y-5→S
:Y-25→N
:Line(245,Y,255,Y,YELLOW)
```

```
:randInt(30,120)→Y
:Y-10→Z
:Y-5→S
:Y-25→N
:Line(245,Y,255,Y,YELLOW)
:Line(245,Z,255,Z,YELLOW)
:Line(240,S,245,Y,YELLOW)
:Line(240,S,245,Z,YELLOW)
:Line(255,Y,260,S,YELLOW)
:Line(255,Z,260,S,YELLOW)
:Line(240,Z,250,N,DARKGRAY
)
:Line(245,Z,240,N,DARKGRAY
)
:Line(245,Z,245,N,DARKGRAY
)
:Line(260,Z,260,N,DARKGRAY
)
:Line(260,S,255,N,DARKGRAY
)
:Line(255,Z,250,N,DARKGRAY
)
:Line(250,Z,245,N,DARKGRAY
)
:Line(250,Z,255,N,DARKGRAY
)
:"GAUGE
:Line(10,12,10,112,BLACK)
:Line(20,12,20,112,BLACK)
:Line(10,112,20,112,BLACK)

:30→F
:For(A,1,F)
:Line(12,A+12,19,A+12,MAGE
NTA)
:End
:For(A,F+1,99)
:Line(12,A+12,19,A+12,WHIT
E)
:End
:"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
:For(A,θ,0,-1)
:Line(65,95,65+30*cos(A°),
95+30*sin(A°),C)
:End
:Text(1,1,"SCORE=",H,"/",T
)
```

```
:End
:
:Degree
:FnOff
:PlotsOff
:1→Xmin
:265→Xmax
:1→Ymin
:165→Ymax
:BackgroundOn Image7
:DispGraph
:ClrDraw
:Text(00,150,"WINNER")
:For(I,0,7)
:Circle(80+I*4,180-(I-10)²
,1,NAVY)
:Circle(80-I*4,180-(I-10)²
,1,NAVY)
:Circle(80+I*10,180-(I-10)
²,1,LTBLUE)
:Circle(80-I*10,180-(I-10)
²,1,LTBLUE)
:Circle(80-I*25,180-(I-10)
²,1,WHITE)
:Circle(80+I*25,180-(I-10)
²,1,WHITE)
:End
:ClrDraw
:Text(10,10,"WINNER")
:For(I,0,10)
:Circle(180+I*4,80-(I-10)²
,1,RED)
:Circle(180-I*4,80-(I-10)²
,1,RED)
:Circle(180+I*10,80-(I-10)
²,1,ORANGE)
:Circle(180-I*10,80-(I-10)
²,1,ORANGE)
:Circle(180-I*25,80-(I-10)
²,1,RED)
:Circle(180+I*25,80-(I-10)
²,1,RED)
:End
:
:ClrDraw
:Circle(180-I*10,80-(I-10)
²,1,ORANGE)
:Circle(180-I*25,80-(I-10)
²,1,RED)
:Circle(180+I*25,80-(I-10)
²,1,RED)
:End
:
:ClrDraw
```

```
)²,1,MAGENTA)
:Circle(120-I*25,160-(I-10
)²,1,MAGENTA)
:Circle(120+I*25,160-(I-10
)²,1,MAGENTA)
:End
:
:ClrDraw
:BackgroundOn Image9
```