

Parabolic Hoops

How's your jump shot? Can you make a basket from anywhere within reasonable range of the hoop? You can with mathematics!! It's as easy as $y = a(x - h)^2 + k$.

In this project, you will create a virtual basketball court. With each new shot, your program will randomly place you, the shooter, a random distance from a 10 ft hoop. The game will generate a reasonable (x,y) location for the maximum point of your parabolic shot. Using the location of the hoop and the vertex, you the shooter will calculate the amplitude, a, needed to "Swish" nothing but net!

Objectives:

Programming Objectives:

- Use the randint() function to generate random integers.
- Use the draw library to draw circles and line segments
- Use loops to repeat code
- Use if statements to make selections

Math Objectives:

- Use the vertex form of a quadratics to solve problems
- Use the absolute value function to validate solutions.

Math Course Connections: Algebra 1 or Algebra 2- Vertex form of a parabola and absolute value

Your project will:

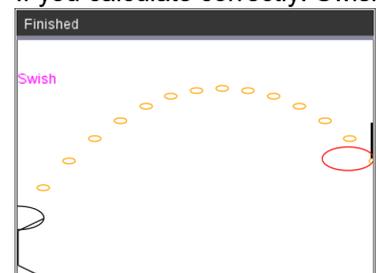
Generate a generate the scene.

```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 8 , 16
Hoop: 13.91 , 10
You: 0 , 5
Enter the amplitude:
```

You will enter the amplitude for the shot.

```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 8 , 16
Hoop: 13.91 , 10
You: 0 , 5
Enter the amplitude: -6/5.91**2
```

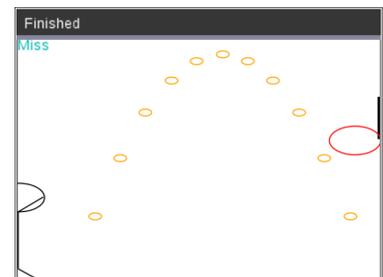
If you calculate correctly: Swish!!



Calculate Incorrectly...Miss... "air ball"

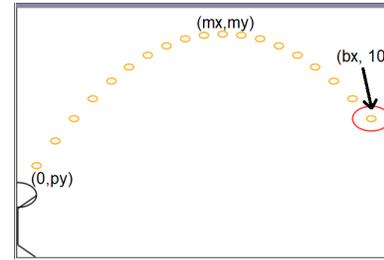
```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 8 , 16
Hoop: 14.2 , 10
You: 0 , 6
Enter the amplitude: |
```

```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 8 , 16
Hoop: 14.2 , 10
You: 0 , 6
Enter the amplitude: -8/4.2**2
```



- The path of your basketball will be parabolic.

That means, it can be written in the form $y = a(x - h)^2 + k$.

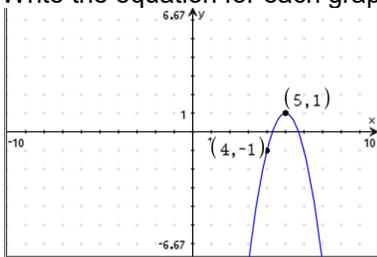


- Let's review the properties for the vertex form of a parabola, $y = a(x - h)^2 + k$.

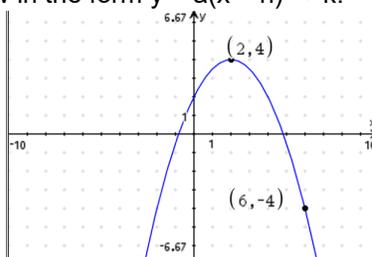
How does a effect the graph of a parabola?

How do h and k effect the graph of a parabola?

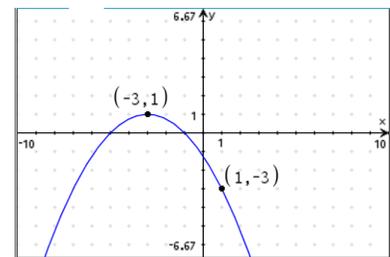
Write the equation for each graph below in the form $y = a(x - h)^2 + k$.



y =



y =



y =

- Graph each of your equations above on a graphs page. Verify your equation contains the points in the diagrams.

Teacher Tip:

a is the amplitude for the parabola. It stretches the graph.
 (h,k) is the vertex of the graph.

$y = -2(x - 5)^2 + 1$

$y = -0.5(x - 2)^2 + 4$

$y = -0.25(x + 3)^2 + 1$

2. The first step in the coding project will be to create a Python random simulation document.

Create a new python project named “basketball”.

Select “Random Simulations” from the Type Menu.

This will automatically import the random library.

You need the randint function from this library to generate random integers.

This also imports the math library.

You will use the fabs function from the math library to find the absolute value.

3. You will need the **time** library.

Place your cursor on the line below the from random import *

Menu> More Modules > Time > **time import ***

This library provides the sleep function.

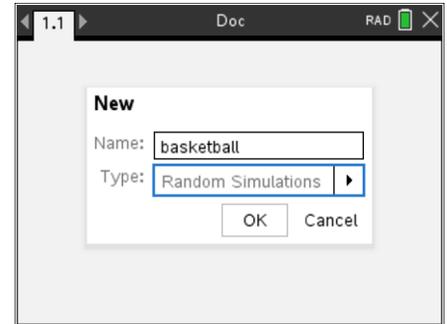
You will use this to add animation to your shot.

`#basket center (bx, by)`

`bx = 0`

`by = 10`

*The line **#basket center(bx, by)** is known as a comment. Programmers use comments to document parts of their code. You may omit this line if you like. Comments start with a # symbol. Notice the comment appears in a light gray color.



For now, let's say the ball will leave the player's hand from either 5 or 6 feet off the ground. The player's x value will be 0.

```
px = 0
py = randint(5, 6)
```

Menu > Random > randint

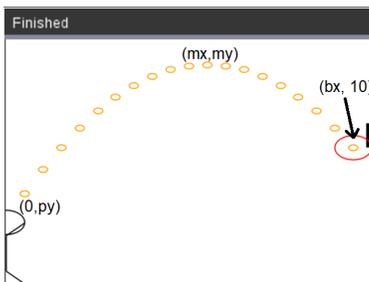
**If you would like the user's shot height to be from the set {5, 5.1, 5.2, ..., 6}
Type `py = randint(50,60)/10`



```
*basketball.py 12/57
from random import *
=====
from time import *

#basket center (bx, by)
bx = 0
by = 10

#player
px = 0
py = randint(5,6)
```



The picture to the left shows the set up for the player, the hoop, and the path for the ball.

The player, will have the point (0, py) where py is the height the ball is released.

The hoop will be regulation in height. Therefore, it will have a height of 10 ft.
You will calculate the x value, bx, the distance to the basketball hoop.

The vertex for the parabola (mx, my) will have a height, my. It will be higher than 10 ft, the height of the hoop. The x value, mx, will be between 0 and bx.

6. The maximum height, **my**, should be over 10 feet to keep it above the rim.
To keep the shot with realistic values, let my be a random integer between 11 and 16 feet.

```
my = randint(11,16)
```

The typical free throw line is 15 feet from the basket.

Assume the max height of the toss occurs anywhere from 3 to 10 feet away from the player.

```
mx = randint(3,10)
```

Now that you have py, mx, and my; where is bx? To answer this question, you must find a, the amplitude first.
Use $y = a(x - h)^2 + k$ to find an equation for a.

7. Did you say: $y = a(x - h)^2 + k$

$$py = a(0 - mx)^2 + my$$

$$py - my = a(mx^2)$$

$$\frac{py - my}{mx^2} = a$$

Add the three lines:

$$my = \text{randint}(11, 16)$$

$$mx = \text{randint}(3, 10)$$

$$a = (py - my) / (mx^{**2})$$

Menu > Random > randint

*In python, **2 is used to square a number.

Press the key [x²] to get **2.

```
*basketball.py 18/57
#basket center (bx, by)
bx = 0
by = 10

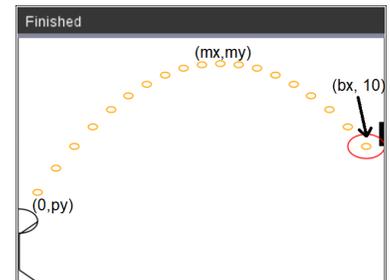
#player
px = 0
py = randint(5,6)

my = randint(11,16)
mx = randint(3,10)
a = (py - my) / (mx**2)
```

Teacher Tip:

Incorrect parenthesis use is a common mistake. Ensure student calculate a correctly with parenthesis.

8. Now that you have py, mx, my and a how can you find the value of bx?



9. Did you say use the formula $y = a(x - h)^2 + k$?

The vertex will be (mx, my). Substitute these values into the appropriate place in the formula.

The point (bx, 10) is one (x, y) point on the equation. Substitute these values into the appropriate place in the formula.

10. Solve the equation for bx: $10 = a(bx - mx)^2 + my$

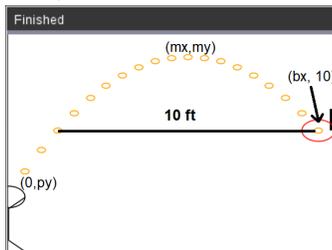
11. Did you get: $10 - my = a(bx - mx)^2$

$$\frac{10-my}{a} = (bx - mx)^2$$

$$\pm \sqrt{\frac{10-my}{a}} = bx - mx$$

$$mx \pm \sqrt{\frac{10-my}{a}} = bx$$

12. The parabola has to x-values that reach a height of 10 feet.



You want the second time the ball reaches 10.

Use the + not the – sign in your equation.

$$bx = mx + \text{sqrt}((10 - my) / a)$$

Menu > Math > Sqrt

```

basketball.py 20/20
by = 0

#player
px = 0
py = randint(5,6)

my = randint(11,16)
mx = randint(3,10)
a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)
    
```

13. Execture your code. Ctrl r

You should get the screen to the right. If not, fix your errors.

If you reach the page on the right, press ctrl ◀ to get back to your code.

```

Python Shell 3/3
>>>#Running basketball.py
>>>from basketball import *
>>>
    
```

14. Now to display the data points for the shot. Add the lines:

```

print("Max", mx, ",", my)
print("Hoop", bx, ",", 10)
print("You", 0, ",", py)
    
```

**print

Menu > Built-Ins > I/O

The words "Max:", "Hoop:", "You:", and "," should all be green because they are string. Everything else should have black font.

```

*basketball.py 26/27
a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)

#Data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
    
```

Execute your code. Ctrl r.

Does it display your data?

The sample on the right is only one possible solution.

```

Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 10 , 15
Hoop: 17.4535599249993 , 10
You: 0 , 6
>>>
    
```

15. To simplify the display, round the hoop's x location to 2 decimal places.

Add the line: `bx = round(bx,2)`
before the print statements.

Menu > Built-Ins > Type > round

Execute your code. Ctrl r.
Verify your display now rounds to two places.

```
*basketball.py 26/27
a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)
bx = round(bx,2)
#Data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
```

16. Now add a line that lets the user enter a guess for the amplitude.
By itself, input lets the user enter a string value. To evaluate the input and save it as a float number, you must `float(eval(input`.

Type:
`pa = float(eval(input("Enter the amplitude: ")))`

Menu > Built-Ins > Type > float
Menu > Built-Ins > I/O > eval
Menu > Built-Ins > I/O > input

```
*basketball.py 27/28
a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)
bx = round(bx,2)
#Data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
pa=float(eval(input("Enter the amplitude: ")))
```

17. To draw everything to the board, import the `ti_draw` library

Menu > More Modules > TI Draw > `from ti_draw import`

```
*basketball.py 29/31
bx = mx + sqrt((10-my)/a)
bx = round(bx,2)
#Data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
pa=float(eval(input("Enter the amplitude: ")))

from ti_draw import *
```

Teacher Tip:

The `ti_draw` must be imported here, not with the other import statements. To see the print statements before python switches to the draw window, the print statements must go before the `ti_draw` import.

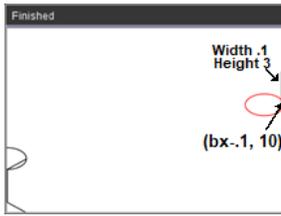
Let the domain be `[0, bx]` and the range `[0, my+ randint(1,5)]`
The `randint` will give some extra cushion in the window.

`set_window(0, bx, 0, my+randint(1,5))`

```
*basketball.py 30/32
bx = mx + sqrt((10-my)/a)
bx = round(bx,2)
#Data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
pa=float(eval(input("Enter the amplitude: ")))

from ti_draw import *
set_window(0,bx,0,my+randint(1,5))
```

19. Now to draw the basketball hoop and backboard.



Backboard:

The width of the screen is bx . Therefore, we will start the backboard at $bx-0.1$. The hoop has a height of 10. We will put the lower left corner at $(bx-0.1, 10)$. Let's make the width of the board 1. (Only 0.1 fits on the screen.) The height of the board will be 3 feet.

Therefore, type `fill_rect(bx - 0.1, 10, 1, 3)`.

```
*basketball.py 35/35
print("You: ",0," ",py)
pa=float(eval(input("Enter the amplitude: ")))

from ti_draw import *
set_window(0,bx,0,my+randint(1,5))

#draw the hoop
fill_rect(bx-0.1, 10, 1, 3)
set_color(255,0,0)
draw_circle(bx-1, 10, 1)
set_color(0,0,0)
```

Red Basket:

To make the hoop red, set the rgb color to: 255, 0, 0.

The net should have a radius of 1 foot. That means the lower left corner will be at $(bx - 1, 10)$. Set the color back to black

```
set_color(255, 0, 0)
draw_circle(bx-1, 10, 1)
set_color(0, 0, 0)
```

**All commands can be found using:

Menu > More Modules > Draw > Shape

or

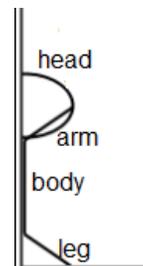
Menu > More Modules > Draw > Control

20. Now, draw the player. For simplicity, your player needs a head, arm, body and leg. The head will be a circle, the arm, leg, and body will be line segments.

The code templates will be:

```
draw_circle(x, y, radius)
draw_line(x1, y1, x2, y2)
```

Remember, the player's height is at (px, py)



What do you think the lines of code will look like? Fill in the templates below with values you think will work for the player.

```
#head
draw_circle( _____, _____, _____)

#body
draw_line( _____, _____, _____, _____)

#arm
draw_line( _____, _____, _____, _____)

#leg
draw_line( _____, _____, _____, _____)
```

Add these lines to the bottom of your code.

Teacher Tip:

One possible solution is given in step 22.

21. Execute your code: ctrl r

The code to the right shows a sample run.

Enter any value for the amplitude and press [enter]

Does your person and basketball hoop look similar to the one on the right?

Run your code several more times. Each time the hoop and player should look a bit different due to the variability in our random variables.

21. One possible solution for the person is shown below.

```
#draw person
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)
```

22. Now to plot the player's shot.

Remember, on about lines 16-21 you coded:

```
my = randint(11,16)
mx = randint(3,10)
a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)
bx = round(bx,2)
```

Answer the following questions:

1. What does the point (mx, my) represent?

3. What does the variable 'bx' represent?

Parabolic Hoops

TEACHER NOTES

Sample Run- Answers will vary

```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 9 , 13
Hoop: 14.51 , 10
You: 0 , 5
Enter the amplitude: |
```

Sample Run- pictures will vary



```
*basketball.py 41/41
fill_rect(bx-0.1, 10, 1, 3)
set_color(255,0,0)
draw_circle(bx-1, 10, 1)
set_color(0,0,0)

#draw person
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)
```

Teacher Tip:

(my, mx) is the vertex for the parabola
a is the amplitude
bx is the distance from the player to the hoop

23. Recall the vertex form for a parabola is: $y = a(x - h)^2 + k$.

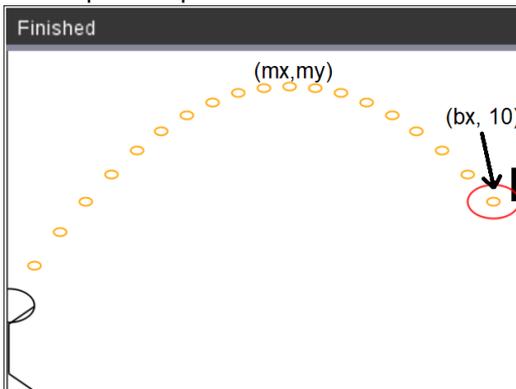
What variables did you use to represent h, the horizontal component for the vertex?

What variables did you use to represent k, the vertical component for the vertex?

Teacher Tip:

h is mx, the x portion of the vertex.
k is my, the y portion of the vertex.

24. To plot the path of the user's shot: You will use the vertex form of a parabola: $y = a(x - h)^2 + k$



Substituting the the point (mx,my) as the vertex:
 $y = a(x - mx)^2 + my$

About line 34 you request the variable pa, as the user's amplitude.
 $y = pa(x - mx)^2 + my$

25. First, change the plot color to orange:

```
#shot
set_color(255,165,0)
```

26. Now to graph the parabolic shot.

Use a loop to cycle through x values starting at 1, ending at bx.

```
for x in range(1, bx+1):
    y=pa*(x - mx)**2 + my
```

Make sure the line $y = pa(x-mx)**2 + my$ has two diamonds to indent the for loop. Python does not use the ^ for exponents. The symbols ** are used for exponents in python.

```
*basketball.py 45/46
set_color(0,0,0)

#draw person
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)

#shot
set_color(255,165,0)
```

```
*basketball.py 46/46
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)

#shot
set_color(255,165,0)
for x in range(1, bx+1):
    y=pa*(x - mx)**2 + my
```

27. Draw circles to represent the path of the ball. The circles will use the x and y values from the for loop. Make the radius 0.25.
`draw_circle(x, y, 0.25)`

Make sure the `draw_circle` has two diamonds in front of the line. This keeps the code part of the for loop.

28. To animate your shot add the line
`sleep(.1)`

29. Execute your program. `ctrl r`

Here is the sample to the right worked out.

$$y = a(x - h)^2 + k$$

$$y = a(x - mx)^2 + my \quad \text{\#max height (mx, my)}$$

$$y = a(x - 10)^2 + 11$$

$$10 = a(14.08 - 10)^2 + 11 \quad \text{\#Use hoop to find the amplitude}$$

$$-1 = a(4.08)^2$$

$$a = -1/(4.08)^2$$

Parabolic Hoops

TEACHER NOTES

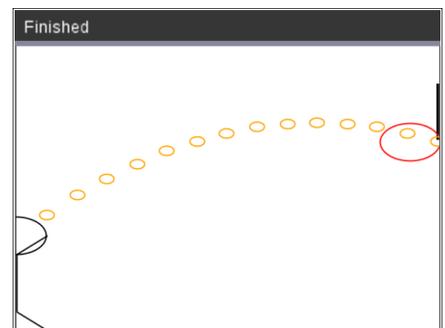
```
*basketball.py 47/47
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)

#shot
set_color(255,165,0)
for x in range(1, bx+1):
    y=pa*(x - mx)**2 + my
    draw_circle(x, y, 0.25)
```

```
*basketball.py 48/48
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)

#shot
set_color(255,165,0)
for x in range(1, bx+1):
    y=pa*(x - mx)**2 + my
    draw_circle(x, y, 0.25)
    sleep(.1)
```

```
Python Shell 6/6
>>>#Running basketball.py
>>>from basketball import *
Max: 10 , 11
Hoop: 14.08 , 10
You: 0 , 5
Enter the amplitude:
```



```
Python Shell 7/7
>>>#Running basketball.py
>>>from basketball import *
Max: 10 , 11
Hoop: 14.08 , 10
You: 0 , 5
Enter the amplitude: -1/4.08**2
>>>|
```

30. Lastly, add code to check the accuracy.

If the difference between the user's entered amplitude and the real amplitude is within 0.2, print "Swish" otherwise print "Miss".

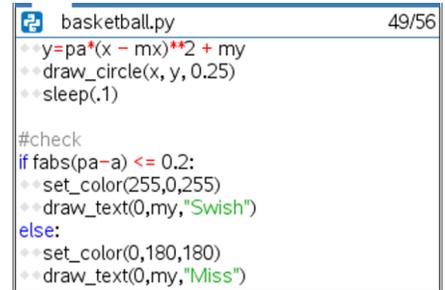
Mathematically, that would be $|pa - a| \leq 0.2$.

Using python, that is `fabs(pa - a) <= 0.2`.

```
if fabs(pa - a) <= 0.2:
    set_color(255,0,255)
    draw_text(0, my, "Swish")
else:
    set_color(0,180,180)
    draw_text(0, my, "Miss")
```

Make sure the `set_color` and `draw_text` have two diamonds for indentation.

- Menu > Built-ins > Control > if..else
- Menu > More Modules> Draw > Control > set_color
- Menu > More Modules > Draw > Shape > draw_text



```
basketball.py 49/56
**y=pa*(x - mx)**2 + my
**draw_circle(x, y, 0.25)
**sleep(.1)

#check
if fabs(pa-a) <= 0.2:
    **set_color(255,0,255)
    **draw_text(0,my,"Swish")
else:
    **set_color(0,180,180)
    **draw_text(0,my,"Miss")
```

31. How many shots can you make in a row before you miss?

Teacher Notes:

Random Simulations

#=====

from math import *

from random import *

#=====

from time import *

#basket center (bx, by)

bx = 0

by = 10

#player

px = 0

py = randint(5,6)

my = randint(11,16)

mx = randint(3,10)

a = (py - my) / (mx**2)

bx = mx + sqrt((10-my)/a)

bx = round(bx,2)

```
#data display
print("Max: ",mx,",",my)
print("Hoop: ",bx,",",10)
print("You: ",0,",",py)
pa=float(eval(input("Enter the amplitude: ")))

from ti_draw import *
set_window(0,bx,0,my+randint(1,5))

#draw the hoop
fill_rect(bx-0.1, 10, 1, 3)
set_color(255,0,0)
draw_circle(bx-1, 10, 1)
set_color(0,0,0)

#draw person
draw_circle(px,py,1)
draw_line(px,1,px,py-1)
draw_line(px,py-1,px+1,py)
draw_line(px,1,px+1,0)

#shot
set_color(255,165,0)
for x in range(1, bx+1):
    y=pa*(x - mx)**2 + my
    draw_circle(x, y, 0.25)
    sleep(.1)

#check
if fabs(pa-a) <= 0.2:
    set_color(255,0,255)
    draw_text(0,my,"Swish")
else:
    set_color(0,180,180)
    draw_text(0,my,"Miss")
```