

Bouncing Ball in Java

Asked 15 years, 9 months ago Modified 11 years ago Viewed 7k times



3



This is probably a really basic problem but I can't seem to find any other articles on it.

Anyway, I have written a small bouncing ball program in Java to try and expand my basic skills. The program is just a simple bouncing ball that will drop and *hopefully* bounce for a while. The original program worked fine but now I have tried to add gravity into the program. The gravity actually works fine for a while but then once the bounces get really small the animation becomes erratic for a very short time then the position of the ball just constantly decreases. I've tried to figure out the problem but I just can't see it. Any help would be most welcome.

```
public final class Ball extends Rectangle {
    float xspeed = 1.0f; float yspeed = 1.0f; float gravity = 0.4f;

    public Ball(float x, float y, float width, float height) {
        super(x, y, width, height);
    }

    public void update(){
        yspeed += gravity;

        move(xspeed, yspeed);

        if(getX() < 0){
            xspeed = 1;
        }
        if(getX() + getWidth() > 320){
            xspeed = -1;
        }
        if(getY() < 0){
            yspeed = 1;
        }
        if(getY() + getHeight() > 200 && yspeed > 0){
            yspeed *= -0.98f;
        }
        if(getY() + getHeight() > 200 && yspeed < 0){
            yspeed *= 0.98f;
        }
    }

    public void move(float x, float y){
        this.setX(getX()+x);
        this.setY(getY()+y);
    }
}
```

}

EDIT: Thanks that seems to have sorted the erratic movement. I'm still struggling to see how I can stop my ball moving down when it has stopped bouncing. Right now it will move stop bouncing then continue moving down passed the "floor". I think it's to do with my yspeed += gravity line. I just can't see how I'd go about stopping the movement down.

java

graphics

physics

Share

Improve this question

Follow

edited Dec 2, 2013 at 5:35



[Zaheer Ahmed](#)

28.5k ● 12 ● 76 ● 111

asked Mar 4, 2009 at 2:05



[Steven Skelton](#)

68 ● 1 ● 8

5 Answers

Sorted by:

Highest score (default)





When you do

3

```
yspeed += gravity;
```



you are *assuming* that the ball has space move through a distance $dx = v_i * t + 1/2 (-g) t^2$. When you are very near the floor this may not be true. It fail if:



- You are near enough the the floor and moving down
- You are very near the floor and have low velocity (like when the ball has lost most of it's energy)



This bug causes your simulation to *stop conserving energy*, resulting in the erratic behavior you see at low amplitude.

You can reduce the problem by using smaller time steps, and you can get rid of it outright if you do test computation to notice when you're out of room and to select a safe time step for *that* iteration (i.e. always use your default unless there is a problem, then calculate the best time step).

However, the basic approximation you're using here has other problems as well. Look in any numeric analysis text for a discussion of solving differential equations numerically.

Share

edited Mar 4, 2009 at 2:29

answered Mar 4, 2009 at 2:24

Improve this answer

Follow



dmckee --- ex-moderator
kitten

101k ● 25 ● 146 ● 235



1

First things first: setting y-speed to 1 when you bounce on the top of the window is not correct, you should set yspeed to -yspeed (but if you start within the borders, it should never bounce up to the top anyway).



Secondly, your multiply by -0.981 when bouncing on the bottom is okay but I'm concerned with the constant 0.4 gravity being added to yspeed every iteration. I think that's what is causing you wiggles at the bottom since you do the move before checking which can result in the ball dropping below ground level.



I would try ensuring the the y value can never go below ground level by replacing the move with:

```
if (getY() + getHeight() + yspeed > 200) {
    move(xspeed, 200 - getY() - getHeight());
} else {
```

```
    move(xspeed, yspeed);
}
```

Share Improve this answer Follow

answered Mar 4, 2009 at 2:25



paxdiablo

880k ● 241 ● 1.6k ● 2k



1

The problem is that when the bounces get really small, the

```
yspeed *= -0.981;
```



line will get called in short succession. The ball will go below the bottom, start coming back up, but still be below the bottom (because $0.981 < 1.0$) eventually, and it will behave erratically. Here's how you fix it:



```
if(getY() + getHeight() > 200){
    yspeed *= -0.981;
    setY(400 - getY() - getHeight()); // I believe this is right.
}
```

By fixing the position, you won't alternate between decreasing and increasing as quickly and won't get stuck in the situation where it is always decreasing because it is always below the bounds.

Share Improve this answer Follow

answered Mar 4, 2009 at 2:25



qpingu

960 ● 7 ● 11

$200 - (\text{getY}() + \text{getHeight}() - 200) = 200 - \text{getY}() - \text{getHeight}() + 200 = 400 - \text{getY}() - \text{getHeight}()$ – qpingu Mar 4, 2009 at 5:16



1

[EDIT: I think I misunderstood, so this probably isn't much use :)]

```
if(getY() + getHeight() > 200){
    yspeed *= -0.981;
}
```



You're negating the vertical velocity on every update. I'd probably try handling gravity in update-sized slices. Assuming you're doing 30 updates per second (for 30fps), maybe something like

```
// Define some constants
SecondsPerUpdate = (1.0f / 30);
AccelDueToGravity = 0.981;

if(getY() + getHeight() > 200){
    yspeed -= (AccelDueToGravity * SecondsPerUpdate);
}
```

Share Improve this answer Follow

answered Mar 4, 2009 at 2:25



[ben_h](#)

4,664 ● 1 ● 22 ● 11



1



I suspect it's because when the ball bounces, it will actually be slightly below the "ground", and at low speeds, it won't move back above the ground in one tick - so the next update() will see it still below the ground, and bounce again - but *downwards* this time, so the cycle continues.

You need to move the ball back up to ground level when it bounces, something like this:

```
if(getY() + getHeight() > 200){
    yspeed *= -0.981;
    setY(200 - getHeight());
}
```

Share

edited Mar 4, 2009 at 2:38

answered Mar 4, 2009 at 2:24

Improve this answer

Follow



[Blorgbeard](#)

103k ● 50 ● 235 ● 276

move seems to take deltas, not absolute values, your ball is likely to fly off the edge of the universe :-)

– [paxdiablo](#) Mar 4, 2009 at 2:27