

# Introduction to algorithms

## Additional resource

This is the algorithm referred to in the 'Introduction to Algorithms' video.

This is also available as a text file (Week5 updategame Algorithm) in the downloaded game frameworks's TheGame versions folder. Spot any mistakes or omissions in this algorithm.

If the ball moves down on the screen perform potential paddle collision

    Get actual distance between the mBall and the ball being checked.

        Check if the actual distance is lower than the allowed => collision

        Get the present velocity

        Change the direction of the ball

        Get the velocity after the collision

        using the fraction between the original velocity and present velocity

        to calculate the needed speeds in X and Y to get the original velocity but with the new angle.

Move the ball's X and Y using the speed (pixel/sec)

Check if the ball hits either the left side or the right side of the screen

But only do something if the ball is moving towards that side of the screen

    change the direction of the ball in the X direction

Get distance between red ball and smiley ball

Check if the actual distance is lower than the allowed => collision

    Get the present velocity

## **Begin programming: Build your first mobile game**

Change the direction of the ball

Get the velocity after the collision

using the fraction between the original velocity and present velocity  $\theta$  to calculate the needed speeds in X and Y to get the original velocity but with the new angle.

increase score

Loop through all SadBalls

Perform collisions (if necessary) between SadBall in current position and the red ball

Get actual distance between the mBall and the ball being checked

Check if the actual distance is lower than the allowed => collision

Get the present velocity

Change the direction of the ball

Get the velocity after the collision

using the fraction between the original velocity and present velocity

to calculate the needed speeds in X and Y to get the original velocity but with the new angle.

If the ball goes out of the top of the screen and moves towards the top of the screen

change the direction of the ball in the Y direction

**Begin programming:**  
**Build your first mobile game**

If the ball goes out of the bottom of the screen  
lose the game