Step 1: I've altered the main function to add visual variety in the creation of bricks with different sizes:

Brick brick(REFLECTIVE, 0.5, -0.33, 0.5, 1, 1, 0);

Brick brick2(DESTRUCTABLE, -0.5, 0.33, 0.8, 0, 1, 0);

Brick brick3(DESTRUCTABLE, -0.5, -0.33, 0.1, 0, 1, 1);

Brick brick4(REFLECTIVE, 0, 0, 0.2, 1, 0.2, 0.5);

Step 2: I've modified the speed to go super fast, with the simple modification:

float speed = 0.10;

Step 3: I've made the bricks destructable after 3 hits,

if (brk->hits == 0) {

brk->onoff = OFF;

}

else {

brk->hits = brk->hits--;

After which the break destroys itself.

Step 4: In an attempt to get the program working for ball collision detection, I created a separate function

void CheckCollision(Circle\* circ)

{

if ((x > circ->x - circ->radius && x <= circ->x + circ->radius) && (y > circ->y - circ->radius && y <= circ->y + circ->radius))

{

however I was unable to achieve the desired functionality. This may be due to how I process over the array for collision detecting in main:

world[i].CheckCollision(&brick);

world[i].CheckCollision(&brick2);

world[i].CheckCollision(&brick3);

world[i].CheckCollision(&brick4);

world[i].CheckCollision(&world[0]);