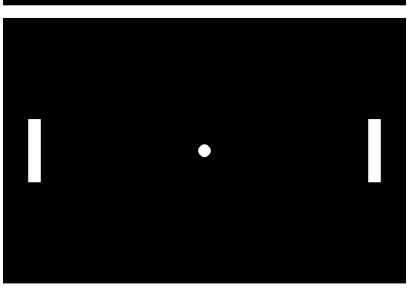
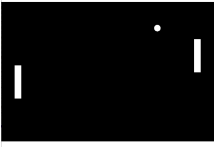



Game Programming: Exercise 5: Pong - Math exercise (2 pages)

Learning objectives	<ul style="list-style-type: none"> • Using the library GLM to solve problems using vectors and matrices • Implementing a matrix, which transforms an object from a local coordinate system to a global (aka. world) coordinate system. <p>Note: When starting the exercise you only see an empty black screen!</p>
Exercise 1	<p>Implement transform</p> <ul style="list-style-type: none"> • Implement <code>Box::getTransform()</code> and <code>Ball::getTransform()</code>. Both methods should create a matrix which transform from the object coordinate frame to the world coordinate frame using translate (position) and scale. • Note that <code>scale.z</code> must be fixed to 0.1f • When implemented correctly the following level should appear: 
<p>Exercise 2</p> 	<p>Move paddles and ball</p> <ul style="list-style-type: none"> • Implement <code>Pong::movePaddle(paddle, yDelta)</code>, where the position of the paddle is moved <code>yDelta</code>. Use <code>glm::clamp</code> to ensure that the paddle does not penetrate the top and bottom bars. • Implement <code>Ball::move()</code>, which should change the ball position based on velocity and delta time.

Exercise 3 	Implement physics <ul style="list-style-type: none"> • Collisions <ul style="list-style-type: none"> o To simulate physics, you need to test if the ball (a circle) collides with an edge (line segment) by implementing the <code>Pong::handleCollision(Edge2D edge)</code>. <ul style="list-style-type: none"> ▪ Hint: Use <code>glm::closestPointOnLine()</code> o If the angle between the edge normal and the ball's velocity is less than 90 degrees, then assume no collision (this solves problems where the ball get stuck in boundary). Hint: Use dot product. • Out of bounds <ul style="list-style-type: none"> o Implement <code>Pong::handleOutOfBounds()</code>: if ball move out of screen increase the score of the other player and relaunch the ball using <code>resetBall(bool)</code>
Exercise 4	Tweak (Extra - just for fun) Customize your game in a fun way. Inspiration: <ul style="list-style-type: none"> • changing gravity • invert controls • dynamic obstacles • and more!
Exercise 5	Convert to sprites (Extra - also just for fun) Use the sprite rendering to render the ball, the paddles and the level.