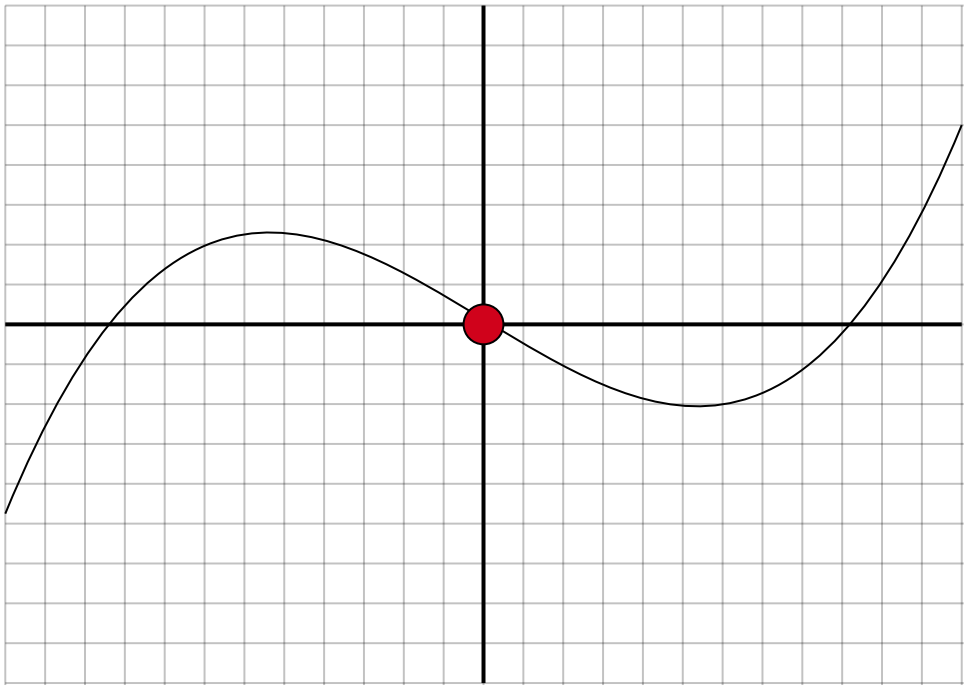
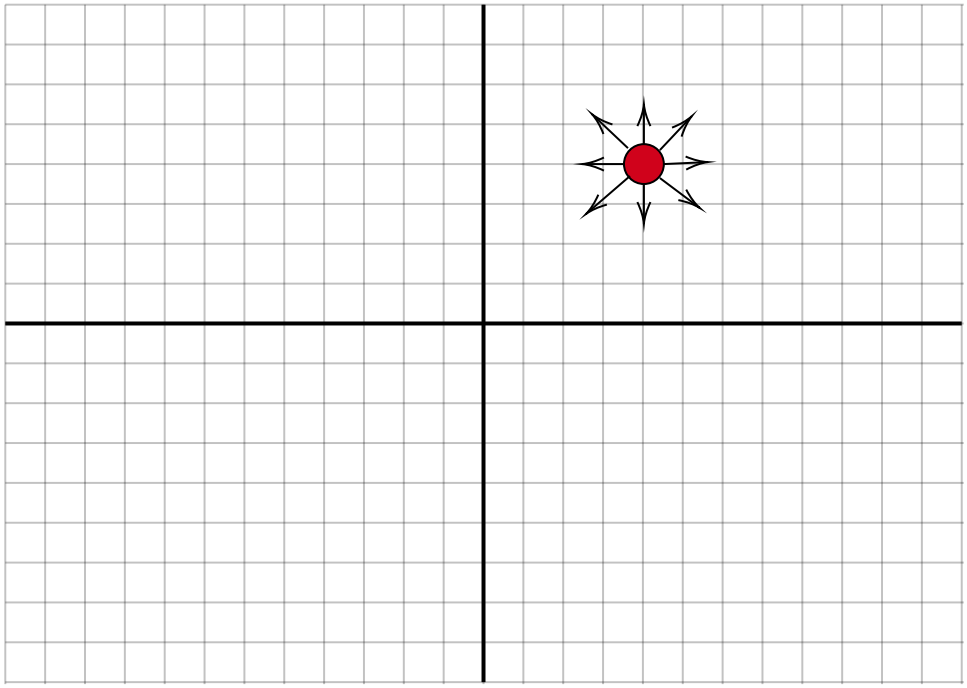


***Problem* 1:** Draw a point at the origin of the canvas. The point should translate to the two corners of the canvas using the sine or cosine function. You should update the position of this point every second.

Use-case: the point starts at (0,0), it will pick a direction of translation (either left or right), the point will also pick an interval of movement per second for example 0.05 units, once the point hits the end of the canvas it should reverse the translation to the other end, animation should continue until the user closes the rendering window, you should not worry about the browser tab being active or inactive.



***Problem* 2:** Draw a circle that moves in a random direction, in x and y direction it should NOT go outside of the canvas, the circle radius should not be more than 0.1, Z-direction should be always equal to zero, your animation should be updated every second.



The point can move to one of it's 8-neighbours, the magnitude depends on the radius of the circle.

$P(x-1, y+1)$	$P(x, y+1)$	$P(x+1, y+1)$
$P(x-1, y)$	$P(x, y)$	$P(x+1, y)$
$P(x-1, y-1)$	$P(x, y-1)$	$P(x+1, y-1)$

***Problem* 3:** For this question, you need to create a triangle in the middle of the CANVAS and use arrow keys to rotate this triangle with respect to the y-axis, if the user presses the left arrow the triangle will rotate left if the user presses the right arrow the triangle will rotate in the opposite direction, the animation should not break your rendering under any condition.

Triangle and implementation properties:

- For rotation, you need to use the Matrix4() class as explained in class