An Interactive, User-driven Physics Simulator

Tom Smilack

TJHSST Computer Systems Lab, 2007-2008

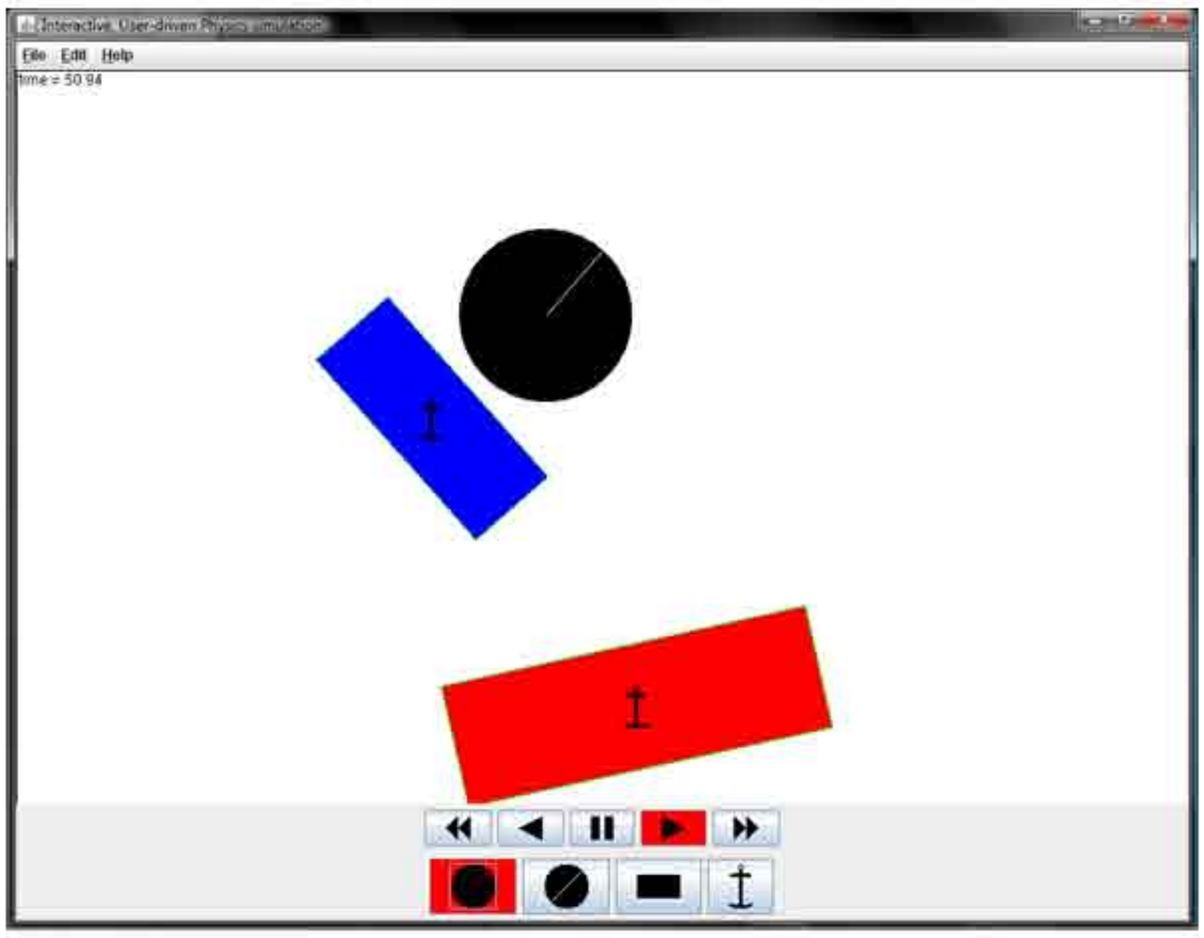
Abstract

Physics simulations are often of single concepts or immune to user control. My project aimed to change that by allowing users to create a situation and then simulating the behavior of objects in that situation. Users can create objects rectangles and circles of any size through intuitive shape tools, then the program converts them to polymorphic objects and runs the simulation. The speed of the simulation can be controlled and users can add more shapes while it is running.

Input

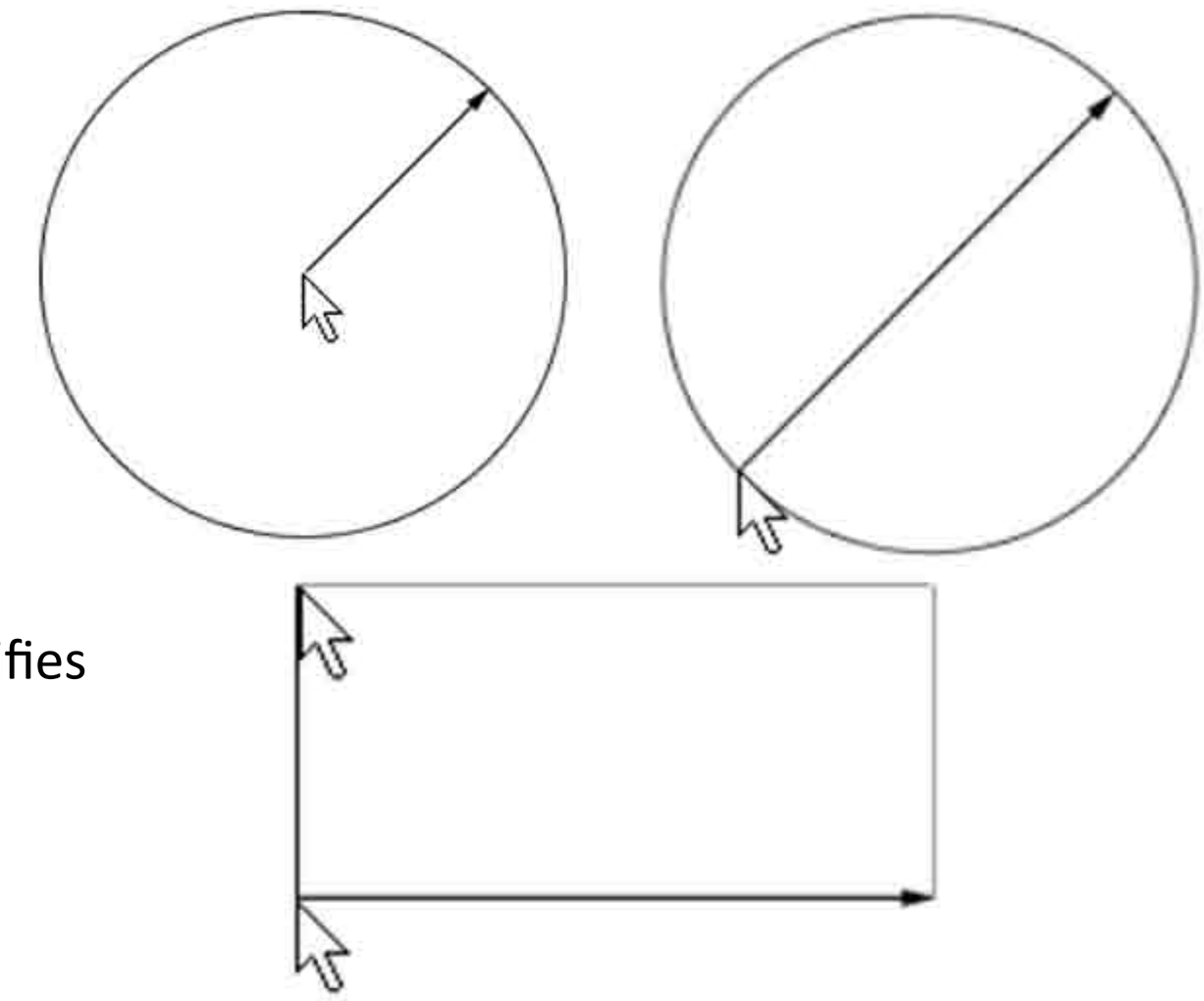
Although users cannot draw in my program, I wanted to employ an intuitive input method. I came up with two methods of creating circles and one of creating rectangles. They are explained in the diagrams to the right. An image of a cursor signifies a click, while a line with an arrow signifies dragging.

Screenshot

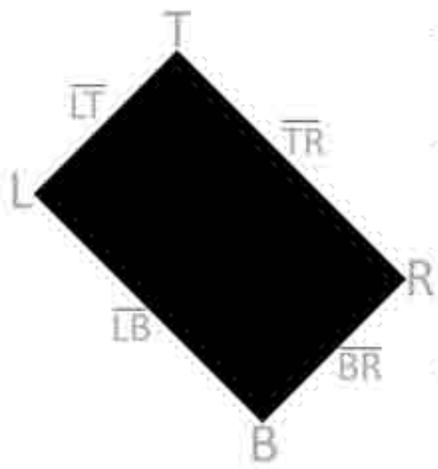


Background

ASSIST, the program that inspired my project, was ade by a team at MIT. In ASSIST, the user uses a "sketchpad" system to draw a situation, which is then interpreted and fed into a commercial physics simulator. My project is similar to ASSIST but focuses more on the physics of the user's situations than on the sketching system.



Rectangle Collision Detection



$$\overline{LT}(x) = (x - L_x) \cdot {}^{(T_y - L_y)} / {}_{(T_x - L_x)} + L_y$$

$$\overline{LB}(x) = (x - L_x) \cdot {}^{(B_y - L_y)} / {}_{(B_x - L_x)} + L_y$$

$$\overline{TR}(x) = (x - T_x) \cdot {}^{(R_y - T_y)} / {}_{(R_x - T_x)} + T_y$$

$$\overline{BR}(x) = (x - B_x) \cdot {}^{(R_y - B_y)} / {}_{(R_x - B_x)} + B_y$$

Results

My program accurately represents projectile motion and collisions with walls without regard to friction, and with an elasticity of one. Rudimentary collisions between moving objects can be seen, but are not completed. Anchors serve their purpose, but the other problems combine to outweigh that. If my program had an accurate way of determining elasticity and friction, then it would seem much more realistic. Problems with collision detection and resolution prevented my project from being successful.