Assignment: A15 (help)

Air-Table: Final features

The following two images highlight the changes that need to be made to the “Spring” class to support the pinned-spring feature.
def collide_on_pucks(self):
    self.plp2_separation_2d_m = self.pl1.pos_2d_m - self.pl2.pos_2d_m

    self.plp2_separation_m = self.plp2_separation_2d_m.length()

    if ((self.plp2_separation_m == 0.0) and (self.length_m == 0.0)):
        spring_force_on_1_2d_N = Vec2D(0.0, 0.0)
    else:
        self.plp2_normalized_2d = self.plp2_separation_2d_m / self.plp2_separation_m

        # Spring force: acts along the separation vector and is proportional to the separation distance.
        spring_force_on_1_2d_N = self.plp2_normalized_2d * (self.length_m - self.plp2_separation_m) * self.strength_Npm

    # Damper force: acts along the separation vector and is proportional to the relative speed.
    v_relative_2d_mps = self.pl1.vel_2d_mps - self.pl2.vel_2d_mps
    v_relative_alongNormal_2d_mps = v_relative_2d_mps.projection_on_to(self.plp2_separation_2d_m)
    damper_force_on_1_2d_N = v_relative_alongNormal_2d_mps * self.damper_Nm2

    # Net force by both spring and damper
    SprDamp_force_2d_N = spring_force_on_1_2d_N - damper_force_on_1_2d_N

    # This force acts in opposite directions for each of the two pucks. Notice the "*" here, this # is an aggregate across all the springs. This aggregate MUST be reset (zeroed) after the movements are # calculated. So by the time you've looped through all the springs, you get the NET force, one each ball, # applied to all individual springs.
    self.pl1.SprDamp_force_2d_N *= SprDamp_force_2d_N
    self.pl2.SprDamp_force_2d_N *= SprDamp_force_2d_N

    # Add in some drag forces if a non-zero drag coef is specified. These are based on the # velocity of the puck (not relative speed as is the case above for damper forces).
    self.pl1.SprDamp_force_2d_N += self.pl1.vel_2d_mps * (-1) * self.drag_c
    self.pl2.SprDamp_force_2d_N += self.pl2.vel_2d_mps * (-1) * self.drag_c