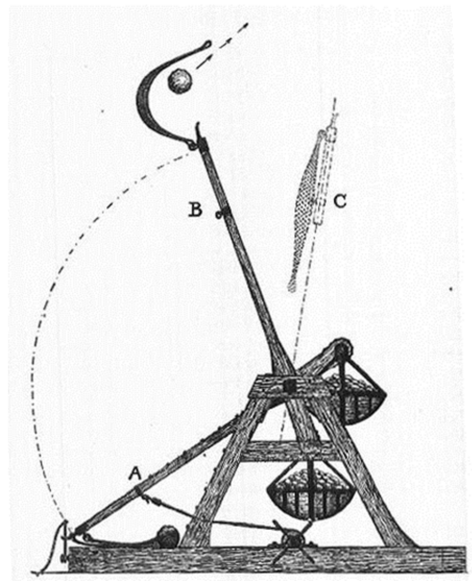
Project 1
Will Drake, Dan Segmiller, and Amy Wood

A trebuchet is a weapon that was commonly used in the Middle Ages. It uses a counterweight on one end of a short arm to launch projectiles at the enemy. This transfer of potential to kinetic energy was once considered the pinnacle of modern weaponry, until the introduction of gun power in the late 13^{th} century.



Trebuchets are designed with many trade-offs in mind. The longer the arm, the farther the projectile will go. But then the base has to be expanded and the device quickly becomes very expensive. Use APM to maximize the range of a given geometry using time of release as the variable. Next, choose several geometries and find the configuration that gives the maximum range per dollar.

Assume you will make your trebuchet out of wood and the items listed below:

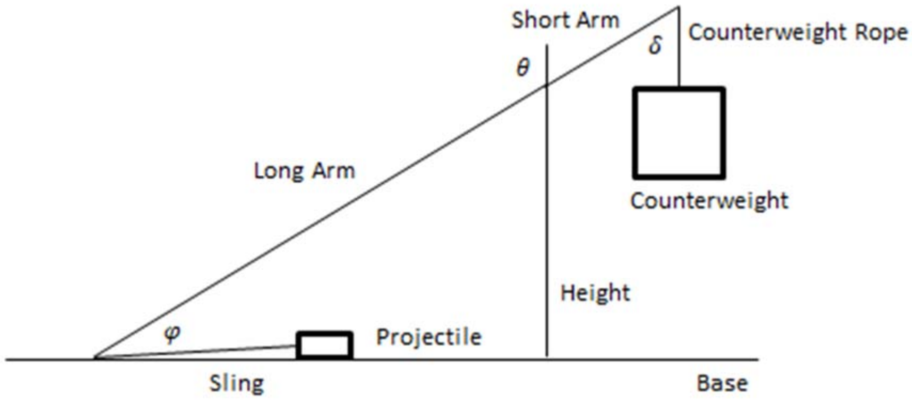
Counterweight	$20
Rope	$3
Fasteners	$10
Rod for Pivot	$7

The total length of wood required can be calculated using:

	$L_{base} = 1.75 * L_5$	
	$L_{width} = 0.75 * L_5$	
	$L_{Total} = 2 * (L_5 + L_{base} + L_{width}) + L_2 + L_1$	

The cost of the wood is:

	$Cost = L_{total} * \$0.45$	



The counterweight has a mass of 100 lb, the projectile has a mass of 1 lb, and the mass of the arms (long and short together) will be 5 lb no matter what their length is optimized to be. The length of the short arm is 1 ft (L1), the length of the long arm is 7.5 ft (L2), the length of the sling is 3 ft (L3), the length of the counterweight rope is 1 ft (L4) and the height of the pivot is 5 ft (L5). The initial condition of the angles can be calculated using the following equations:

	$\varphi = sin^{-1}(\frac{L_5}{L_2})$	
	$\theta = \pi - \varphi$	
	$\delta = \frac{\pi}{2} - \varphi$	

These angles during the launch can be calculated using the equations for δ, θ, φ included in the Appendix A. Once calculated, these angles can be used to calculate the height of the projectile (will in the sling) and the velocity of the projectile in the x and y directions using these equations:

	$y_3 = -(L_3 * \cos(\delta - \theta) - L_2 * \cos(\theta)) + L_5$	
	$v_x = -L_3 * \cos(\delta - \theta) * \left(\frac{d\delta}{dt} - \frac{d\theta}{dt}\right) - L_2 * \cos(\theta - \frac{d\theta}{dt})$	
	$v_y = L_3 * \left(\frac{d\delta}{dt} - \frac{d\theta}{dt}\right) * \sin(\delta - \theta) - L_2 * \frac{d\theta}{dt} * \sin(\theta)$	

After the velocities in the x and y directions are known, the range can be calculated using:

	$range = \frac{2 * v_x * v_y}{g}$	

where g is acceleration due to gravity.

Hint: The optimization will calculate range values for as long as the simulation is allowed to run. The equations, however, are only valid while the projectile is still in the sling. To verify that you aren't using a maximum range value from the non-applicable time after the projectile's "release", plot the height variable (y_3) against time: any maxima past the time where y_3 equals the sum of L2, L3, and L5 are invalid.

Appendix A

During simulation, the values of phi, theta, and delta must be calculated using differential equations. For convenience, those equations are included below.

```
! THETA EQUATION
      0 = ((L1^2 - L1*L2 + L2^2)*mb*d2THE)/3 + (g*(L1 - L2)*mb*s_T)/2 + g*m2*(-(L3*s_PT) -
L2*s_T) - (m2*(2*(-(L3*(d1PSI - d1THE)*c_PT) - L2*d1THE*c_T)*(-(L3*(d1PSI - d1THE)*s_PT) +
L2*d1THE*s_T) +     2*(-(L3*d1PSI*c_PT) +  d1THE*(L3*c_PT - 2*c_T))*(L3*d1PSI*s_PT + d1THE*(-
(L3*s_PT) - L2*s_T))))/2 + (m2*(2*(-(L3*(d1PSI - d1THE)*c_PT) -  L2*d1THE*c_T)*  (-(L3*(d1PSI
- d1THE)*s_PT) + L2*d1THE*s_T) + 2*(L3*c_PT - L2*c_T)* (-(L3*(d2PSI - d2THE)*c_PT) -
L2*d2THE*c_T +  L3*(d1PSI - d1THE)^2*s_PT + L2*d1THE^2*s_T) + 2*(-(L3*(d1PSI - d1THE)*c_PT) -
L2*d1THE*c_T)*  (L3*d1PSI*s_PT + d1THE*(-(L3*s_PT) - L2*s_T)) + 2*(-(L3*s_PT) - L2*s_T)*
(L3*d1PSI*(d1PSI - d1THE)*c_PT +  d1THE*(-(L3*(d1PSI - d1THE)*c_PT) - L2*d1THE*c_T) +
L3*d2PSI*s_PT +  d2THE*(-(L3*s_PT) - L2*s_T))))/2. +  g*m1*(L1*s_T - L4*sin(PHI + THE)) -
(m1*(2*(L1*d1THE*c_T - L4*(d1PHI + d1THE)*cos(PHI + THE))* (L1*d1THE*s_T - L4*(d1PHI +
d1THE)*sin(PHI + THE)) + 2*(L1*d1THE*c_T - L4*(d1PHI + d1THE)*cos(PHI + THE))* (-
(L1*d1THE*s_T) + L4*(d1PHI + d1THE)*sin(PHI + THE))))/2. + (m1*(2*(L1*d1THE*c_T - L4*(d1PHI +
d1THE)*cos(PHI + THE))* (L1*d1THE*s_T - L4*(d1PHI + d1THE)*sin(PHI + THE)) + 2*(L1*d1THE*c_T
- L4*(d1PHI + d1THE)*cos(PHI + THE))*(-(L1*d1THE*s_T) + L4*(d1PHI + d1THE)*sin(PHI + THE)) +
2*(L1*c_T - L4*cos(PHI + THE))* (L1*d2THE*c_T - L4*(d2PHI + d2THE)*cos(PHI + THE) -
      1*d1THE^2*s_T + L4*(d1PHI + d1THE)^2*sin(PHI + THE)) + 2*(L1*s_T - L4*sin(PHI +
THE))*(L1*d1THE^2*c_T - L4*(d1PHI + d1THE)^2*cos(PHI + THE) + L1*d2THE*s_T - L4*(d2PHI +
d2THE)*sin(PHI + THE))))/2.0

! PHI EQUATION
    0 = -(g*L4*m1*sin(PHI + THE)) - (m1*(2*L4*(d1PHI + d1THE)*(L1*d1THE*c_T - L4*(d1PHI +
d1THE)*cos(PHI + THE))* sin(PHI + THE) - 2*L4*(d1PHI + d1THE)*cos(PHI + THE)*(L1*d1THE*s_T -
L4*(d1PHI + d1THE)*sin(PHI + THE))))/2.0+ (m1*(2*L4*(d1PHI + d1THE)*(L1*d1THE*c_T - L4*(d1PHI
+ d1THE)*cos(PHI + THE))*  sin(PHI + THE) - 2*L4*(d1PHI + d1THE)*cos(PHI + THE)*
(L1*d1THE*s_T - L4*(d1PHI + d1THE)*sin(PHI + THE)) - 2*L4*cos(PHI + THE)*(L1*d2THE*c_T -
L4*(d2PHI + d2THE)*cos(PHI + THE) - L1*d1THE^2*s_T + L4*(d1PHI + d1THE)^2*sin(PHI + THE)) -
2*L4*sin(PHI + THE)*(L1*d1THE^2*c_T - L4*(d1PHI + d1THE)^2*cos(PHI + THE) + L1*d2THE*s_T -
L4*(d2PHI + d2THE)*sin(PHI + THE))))/2.0

! PSI EQUATION
      0 = g*L3*m2*s_PT - (m2*(2*L3*(d1PSI - d1THE)*(-(L3*(d1PSI - d1THE)*c_PT) -
L2*d1THE*c_T)*s_PT + 2*(L3*d1PSI*c_PT - L3*d1THE*c_PT)* (L3*d1PSI*s_PT + d1THE*(-(L3*s_PT) -
L2*s_T))))/2.0+ (m2*(2*L3*(d1PSI - d1THE)*(-(L3*(d1PSI - d1THE)*c_PT) - L2*d1THE*c_T)* s_PT -
2*L3*c_PT*(-(L3*(d2PSI - d2THE)*c_PT) - L2*d2THE*c_T + L3*(d1PSI - d1THE)^2*s_PT +
L2*d1THE^2*s_T) + 2*L3*(d1PSI - d1THE)*c_PT*(L3*d1PSI*s_PT + d1THE*(-(L3*s_PT) - L2*s_T)) +
2*L3*s_PT*(L3*d1PSI*(d1PSI - d1THE)*c_PT + d1THE*(-(L3*(d1PSI - d1THE)*c_PT) - L2*d1THE*c_T)
+ L3*d2PSI*s_PT + d2THE*(-(L3*s_PT) - L2*s_T))))/2.0
```