

### Program 0: Due 9/3

A track coach wants to predict where her runners are on a 600 foot course. One complication is that the course is an out-and-back course, so runners must stop and turn at the half way point. Write a program that allows her to specify each runner's name, top speed (m/h) and acceleration (f/s/s). Print the predicted location (distance from the start) of each runner every 5 seconds. There is a maximum of 5 runners.

Your output should be similar to the following table:

Runner	Max Speed (f/s)	Acceleration (f/s/s)
Nelly	30.0	8.0
Steve	8.8	3.0

  

Time	Nelly	Steve
0.0	0.0	0.0
10.0	243.8	75.1
....		
xx.x	0.0	0.0

To correctly model the distance of each runner, you must correctly model the acceleration of each runner. Recall that  $\text{distance} = \text{velocity} * \text{time}$ , and  $\text{velocity} = \max(\text{acceleration} * \text{time}, \text{max\_speed})$ . At the start of the race, each runner initially has zero velocity and can only speed up to top speed at the acceleration specified. At the turn around, each runner must decelerate at the acceleration specified, and then accelerate in the return direction.

Include Nelly, Steve, and Usain (max speed = 41.0 f/s, acceleration = 11.0 f/s/s) in the output you generate.

Create a class Runner. Create an instance of this class for each runner the coach enters. Your main in the Runner class should produce the output for the three runners given above. (That is, I should be able to compile and run your Runner.java code, and without further input, the table for three runners will be generated.)

One way to approach this problem is to iterate over some suitably small time interval (say 0.01 seconds), computing the velocity and distance for the interval, and reporting the distance each 10 seconds. You must remember to decelerate your runners as they approach the turn around.

