

Homework 1: In particle simulations, as well as in games, there is typically a loop that represents the passage of time. Each time this loop runs, the positions of things (particles, characters, bullets, etc.) must have their positions updated so that they can be drawn appropriately. The following code represents a rudimentary method which is meant to update the positions of things/particles (represented by the coords array which holds x,y,z coordinates for each thing/particle) based on their velocities (which holds speeds in x,y,z for each thing/particle). This is the sort of method that might be called by the time loop before redrawing the screen. Do a detailed analysis of this code, show me how you arrived at your answer (show your math!), then discard extraneous terms and constants to provide me with a $O(f(N))$ estimate of the time complexity.

```
1 /**
2  @param coords a 2 dimensional array of doubles - the first dimension is the number of particles, and the second di
3  @param velocities a 2 dimensional array of doubles - the first dimension is the number of particles, and the secon
4  @param numParticles - an integer representing the number of particles in the coords and velocities arrays
5  @param timeStep a double representint the amount of "virtual" time (in seconds) that will have passed
6 */
7
8 public void updatePositions(double[][] coords, double[][] velocities, int numParticles, double timeStep) {
9
10     for (int index=0; index < numParticles; index++) {
11
12         if (coords[index][1] <= 0.0)
13             continue;
14         else {
15             coords[index][0] = coords[index][0] + (velocities[index][0] / 2.0) * timeStep;
16             coords[index][1] = coords[index][1] + (velocities[index][1] / 2.0) * timeStep;
17             coords[index][2] = coords[index][2] + (velocities[index][2] / 2.0) * timeStep;
18         }
19     }
20
21     velocities[index][1] = velocities[index][1] - 9.8 * timeStep;
22 }
23
24
25 }
```

Submission: Upload your document with your analysis into gitlab, in a subdirectory of your git repository named HW1.