

The Game of Death

You have just joined Racoon Consulting as a Software Analyst, and have been drafted to a pro bono project to analyse the spread of the recent COVID-19 epidemic.

Your first task is to create the proof of concept for an extensible simulation framework that will be used to model the movement of humans within their habitats, and to generate useful statistics on how the epidemic is spread with varying human behavior.

A habitat, for the proof of concept, is defined as a 2-D square grid of dimensions $n \times n$ m², where $0 < n \leq 1000000$, with 2D coordinates i and j in the same range for each dimension. One and only one human may occupy any cell (i,j) on the grid. For the proof of concept, you may select random initial locations, called homes, for each human on the grid, upon driver initialization.

A habitat may represent a house, a street, a block or even a city.

There are h humans living in the grid, where $1 < h \leq 1000000$.

Each human has a string name n , which may be the human number (from 1 to h) for the proof of concept, but will eventually contain the names of actual humans.

Each human has an immunity level i which is a number between 0 and 1 inclusive, and represents the probability of the human contracting the virus if the human is in contact with another human already containing the virus. For the PoC, your driver may generate this probability randomly, but eventually, you expect to initialize it by some other means.

Contact is defined as being true if two humans are in adjacent cells, diagonally adjacent cells are included as well.

Of the h humans, w are working humans, who have an office to go to, where $h \geq w \geq 0$.

There are o offices on the grid on random locations, where $0 \leq o \leq 10000$. Each office o has a name n , which may be the office number for the PoC, and is initially unoccupied by any humans. Each of the w humans works at exactly one of these offices, the association is selected at random for the PoC.

You need to design a class hierarchy/framework along with a driver program that would enable the system to generate or load data following the above constraints. You need to read the parameters defined above (n , h , i , w and o) in your driver program at runtime, and generate the random dataset for the PoC based on the parameters by any mechanism of your choice.

Level 2: To increase your chances for success, you can opt to enhance your framework such that a time ticker based simulation may be run. On each tick, the worker humans should move towards their office by moving one cell in either the x or the y axis direction towards the office, or staying in the same place, following the previously defined rules. You can resolve contentions as you see fit. Once the worker human reaches the office, the human should immediately begin traveling back to their home. On each tick, after moving all humans thus, the immunity level for each uninfected human that is in contact with an affected human should be used to determine if the contact results in infecting the first human.

Upon contracting the virus, a count-down timer should start for the human such that after t ticks, the infected human switches back to the uninfected state. The parameter t should also be input as the rest of the parameters and is the same for each human.

NOTE: While designing, keep in mind that the system would need to be enhanced in the future to incorporate more attributes, and also that Racoon Consulting has the ambitious goal to model the actual population of a whole city eventually. Your solution must account for these facts, and would be judged on design elegance, edge case handling, performance considerations while choosing the model, robustness, extensibility and code clarity.

The choice of the language is entirely up to you, but we recommend an object oriented language that allows you to write extensible code.

You will certainly come up with questions that need to be answered, as long as the provided instructions do not answer them, you can send your queries to sohaib.athar@afiniti.com

We expect this test program to be doable in one to two days, it would be ideal if you can finish it by Monday, 13th July, 2020, but if you are busy, you may take up to the 17th July to submit the solution.

The solution should be submitted by sharing a private repository on github with the user sohaibathar@gmail.com. A README.md file must be present at the root of your repository explaining your solution design, an approximate distribution of your time spent on the problem's various phases, how to run it and the thought process behind the choices you've made. The repository should contain multiple commits at different phases of your development so we can see the evolution of the code.

Good Luck!