Python Assessment 2- Software Information Document

# Software development process:

## Basic software functions:

### Intention of software use:

The software is designed to model the resulting location of bacteria in the event of a biological weapon being released from the top of a building. As the bacteria fall they are subjected to wind direction and updraft affecting their final resting place and subsequently the infected area which would need to be quarantined.

### Initial software functioning:

Hence the software reads in an environment read i from a CSV file which identifies the location of a 75 meter tall building, from which the bomb is relseased. In the initial model there is a 20% chance of updraft lifting the bacteria a meter above the building per iteration of the model. There is a 10 % chance the height of the bacteria will not change when it is above the building and a 70% chance it will fall 1 meter per model iteration. Bellow the height of the building the particle is no longer subjected to updraft so falls 1m per iteration until it reaches the ground. Each iteration the bacteria are also affected by wind direction; there is a 10% chance the bacteria particle will travel North per iteration, 10% chance it will move south, 5% chance it will move a meter west and 75% it will move a meter east per iteration. Once the bacteria reach the ground, they can no longer move north south east or west.

The software produces a visual output of the final bacteria locations as a density map and also writes out the density map to an CSV file, where cumulation of bacteria in the same location can be observed: the count of each comma separated variable indicates how many bacteria fell in that location.

## Further software development:

### Visual enhancement:

The 'seaborn-poster' matplotlib style was used to improve the visual aesthetics of the outputs. Furthermore the bacteria marker sizes were decreased in size for clarity and a marker produced for both the start point of the bacteria, a star, and the mean end coordinates of the bacteria, a cross. The mean coordinates indicate the epicentre of the bacteria outbreak, which could be used as the point from which an evacuation zone was decided.

Furthermore, the original graph output lf the bacteria density map was on the same 300x300 grid, as to be easily comparable to the csv written in, however this did not allow the user to see some of the detail of the bacterial dispersal as it was too zoomed out. Therefore, a second visual output was created that uses the maximum and minimum coordinates of the most outlying bacteria to centre the graph to the bacterial dispersal.

### User input:

User input was added to make the software applicable to more situations and allow it to model a wider range of scenarios. The building height, probability of updraft and number of agents was made alterable by the user in the console in Spyder of from the notebook in Jupyter notebook. Originally the software was developed to also allow the user to change the number of iterations, however this could result in outputs that do not show the final location of the bacteria, as too low a number of iterations could capture bacteria which are still airborne. Hence the number of iteration has subsequently been set to be calculated from the entered building height and probability of an updraft, the two factors that affected the number of iterations required for all the bacteria to reach the ground.

## Issues faced during software development:

### User input of wind direction:

It was considered to also allow the software user to alter the probability of the particle being blown north, south, east, or west. However due to the way the probabilities are set it would result in very complex maths and require the user to make sure they entered probabilities that added up to 1 as to not break the code. This issue was overcome for the updraft by noting the user to the input range required for the probability, as there were less variables to consider.

### Creation of an Agent framework:

Originally an agent framework was created however due to the user input resulted in over complication as the two files had to communicate with each other, also resulting in slower running code due to increased memory usage.

### CSV file size:

The original CSV output file of the density map was designed to be 300x300 to be directly comparable to the input file. However due to the user input function not all the agents always fall in the 300x300 grid, i.e. when the building is tall with a high probability of updraft. The csv output size could be increased, however as the building height is infinite so too could the output area be. Therefore the software is designed to produce a CSV output if all the bacterial particles are within the 300x300 grid, if not the CSV file is not produced and a message is shown to the user informing them. However, the visual outputs are still produced for the larger outputs.