Cellular Automata – Hunter and Prey

The goal for this programming project is to create a simple 2D predator–prey simulation implemented as a cellular automata. In this simulation, the prey are Hobbits, and the predators are Nazgul. These beings live in a fictional 2D world composed of a 30 X 30 grid of cells. Only one being may occupy a cell at a time, and the Hobbits are constantly on the run from the voracious Nazgul who like nothing better than to feed on the plump, un-muscled, succulent Hobbit flesh. Around the edge of this world there is an ominous city wall. Due to the casting of some ancient black magic the Nazgul are not capable of crossing this wall. They are essentially repelled and must change direction. On the other hand, the Hobbits, who appeased the witches with hand crafted ales and excellent long bottom leaf, are capable of ascending these walls and tele-porting to the other side of town. When a Hobbit approaches a particular edge it wraps around to the opposite edge and appears there, similar to the mythological beast Pac Man. As this world is an automata, it does not require any human interaction. It is simply seeded with life and set on its inevitable path. As the seeding is random there are infinite scenarios that can play out. Time in this world is simulated in discrete steps which can be implemented as a loop iteration. A single iteration over the entire world is equivalent to a 24-hour Earth day. Each being performs some action every time step. While each species is of the same genetic lineage and share the same basic behaviors (Move and Reproduce), the implementation is vastly different.

Hobbits:
- Move
  - A Hobbit will randomly attempt to move up, down, left, or right. If the cell in the selected direction is occupied, it will try another direction. If all four directions are occupied the Hobbit does not move.
  - A Hobbit can move "across the wall" and end up on the opposite side of the grid.
- Reproduce
  - If a Hobbit survives for three "days," then at the end of the third day (after moving) the Hobbit will asexually reproduce. A new Hobbit will be spawned in an adjacent empty cell (up, down, left, or right). If none of these four cells is empty, no new Hobbit will be produced.
  - Once an offspring is produced, this particular Hobbit cannot produce another until three more time steps have elapsed.

Nazgul:
- Move
  - If there is a Hobbit in an adjacent cell (up, down, left, or right), the Nazgul will move to that cell, stab the Hobbit with its Ringwraith sword, and consume it.
  - If there are no adjacent Hobbits, the Nazgul moves randomly like a Hobbit. Nazgul cannot cross walls to move to the opposite side of the grid.
- Reproduce
  - If a Nazgul survives for eight "days," then at the end of the eighth day (after moving) it will spawn a new Nazgul like the Hobbits do.
- Starve
  - If a Nazgul has not eaten a Hobbit within the last three "days," then at the end of the third day (after moving) it will starve and die. It should then be removed from the grid.

Assignment: Understanding the Automata

To understand the game better, you will play a few rounds on paper on a smaller grid.
1. Draw a square around a 5x5 grid on your graph paper. Label it "Gen 0."
2. Randomly assign starting locations of Nazgul and Hobbits:
    a. Represent Nazgul with an x and Hobbits with a dot.
    b. Roll the 8-sided die you were given for each cell
    c. 1-3 gives an empty cell, 4-7 gives a Hobbit, and 8 gives a Nazgul.

For each round:
1. Enclose another 5x5 grid on your graph paper and label it with your next generation. The first time, it will be "Gen 1."
2. Copy your current grid's contents lightly in pencil.
3. Traverse the board from left to right through each row. If the cell has a creature in it, take the appropriate actions for the creature type.
    a. When moving at random, use the 4-sided die to choose direction – 1 is up, 2 is left, 3 is right, and 4 is down.
    b. A Nazgul will always move to the first Hobbit it notices. It should check each direction in order until it finds a Hobbit and moves to it; if it does not, it should move randomly.
    c. You will need to track the number of moves since reproduction (or since spawn) for all creatures; do this by writing this number as a subscript on the creature's symbol.
    d. Nazgul also need to track number of moves since the last time they ate a Hobbit. Use a superscript for this.

Repeat this process for at least 8 turns, or until one of your populations dies out (whichever comes first).

Questions:
1. Give at least one way that you could simulate the rolling of a d4 to randomly choose a direction in Python.

2. Below are some Hobbit coordinates and a direction they are randomly chosen to move. Assume that the given direction is a free space. What are the new coordinates of the Hobbit after moving?

        (2, 3) right

        (1, 0) down

        (1, 0) left

        (4, 4) down

        (3, 1) up

3. Considering your answers above, consider writing a function for the "random move." How can you update coordinates for the edge case (when the Hobbit teleports to the other side)?