4.

For a square si, let a possible way of getting to it be ai1, ai2, ai3, ai4, ai5.

P(a1), P(a2) =

P(a3), P(a4), P(a5) =

P(si) =

Pseudo Code:

1. Initialise 4 x 3 matrix to all zeros where [x][y] is the probability of
2. seq = {up, up, right, right, right}
3. for all a in sequence,
   1. for b in possible actions, {a, left(a), right(a)},
   2. if a == b, probability of sequence so far \* 0.8
   3. else, probability of sequence so far \* 0.1
4. For all the paths to a specific square, sum them and store them in the matrix

Result (using Python):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
| 1 | 0.025240 | 0.06224 | 0.17994 | 0.32792 |
| 2 | 0.18054 |  | 0.04443 | 0.01368 |
| 3 | 0.024620 | 0.02824 | 0.02627 | 0.08688 |

5.

U(start, up) = =

=

=

U(start, down) = =

=

Pick up when U(start, up) > U(start, down)

Since 0 < ,

Hence, up is chosen when . Conversely, down is chosen when .