## Random Walk Algorithms: Homework 4

- 1. Write code to "blur" a 1D "greyscale image" on  $\mathbb{Z}/128\mathbb{Z}$ , *i.e.*,  $\vec{u}_0 \in \{0, 1, \ldots, 255\}^{128}$ , using  $B^t$  where  $B = (X + 2I + X^{-1})/4$ . That is, pixel s in the "blurred image" is  $\lfloor (B^t \vec{u}_0)_s \rceil \in \{0, 1, \ldots, 255\}$ . Plot two examples to illustrate that your code is working as it should, for different values of t.
- 2. Write code to "unblur" the output of your blurring code, without using the original "image".
  - a. First do it assuming that you know which t was used, *i.e.*, that the image was blurred by  $B^t$  for the correct t.
  - b. Plot the results of unblurring the examples you showed in problem 1 and compare them with the originals.
  - c. Second, suggest a strategy for unblurring if you don't know t.