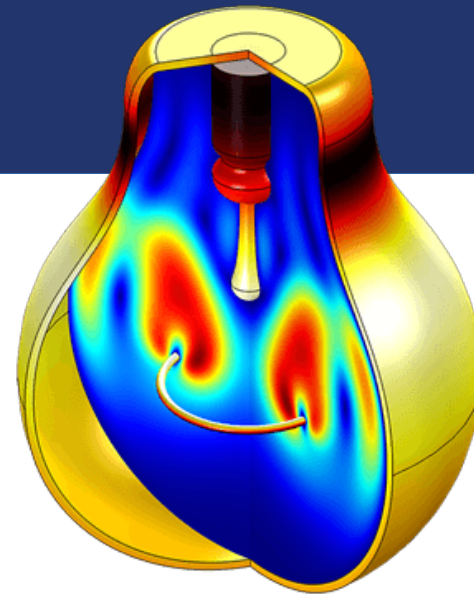


Parallel programming

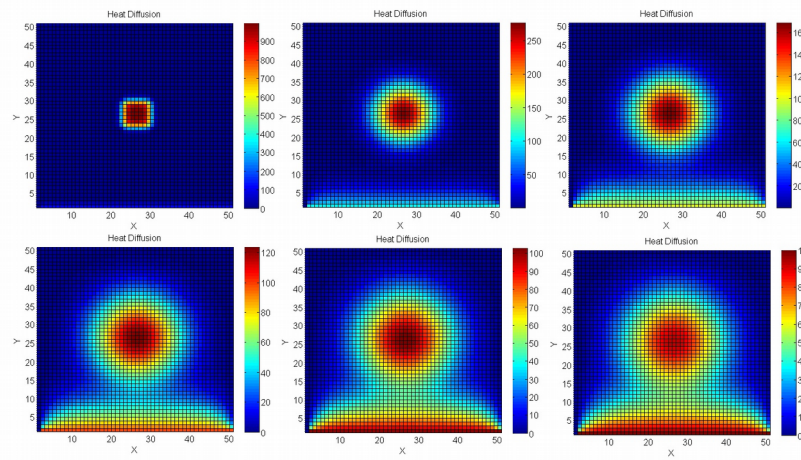
HW3 assignment





What is a heat diffusion

- Heat diffusion/transfer
 - exchange of thermal energy between physical systems
 - rate of heat transfer is dependent on the temperatures of the systems





Simplified model

- Discretization of the space into **cells** (i.e., matrix) with constant temperature
- The new temperature in the given coordinates (i,j) is equal to the average of old temperatures of all 9 spots in neighborhood

$t[i-1][j-1]$	$t[i-1][j]$	$t[i-1][j+1]$
$t[i][j-1]$	$t[i][j]$	$t[i][j+1]$
$t[i+1][j-1]$	$t[i+1][j]$	$t[i+1][j+1]$

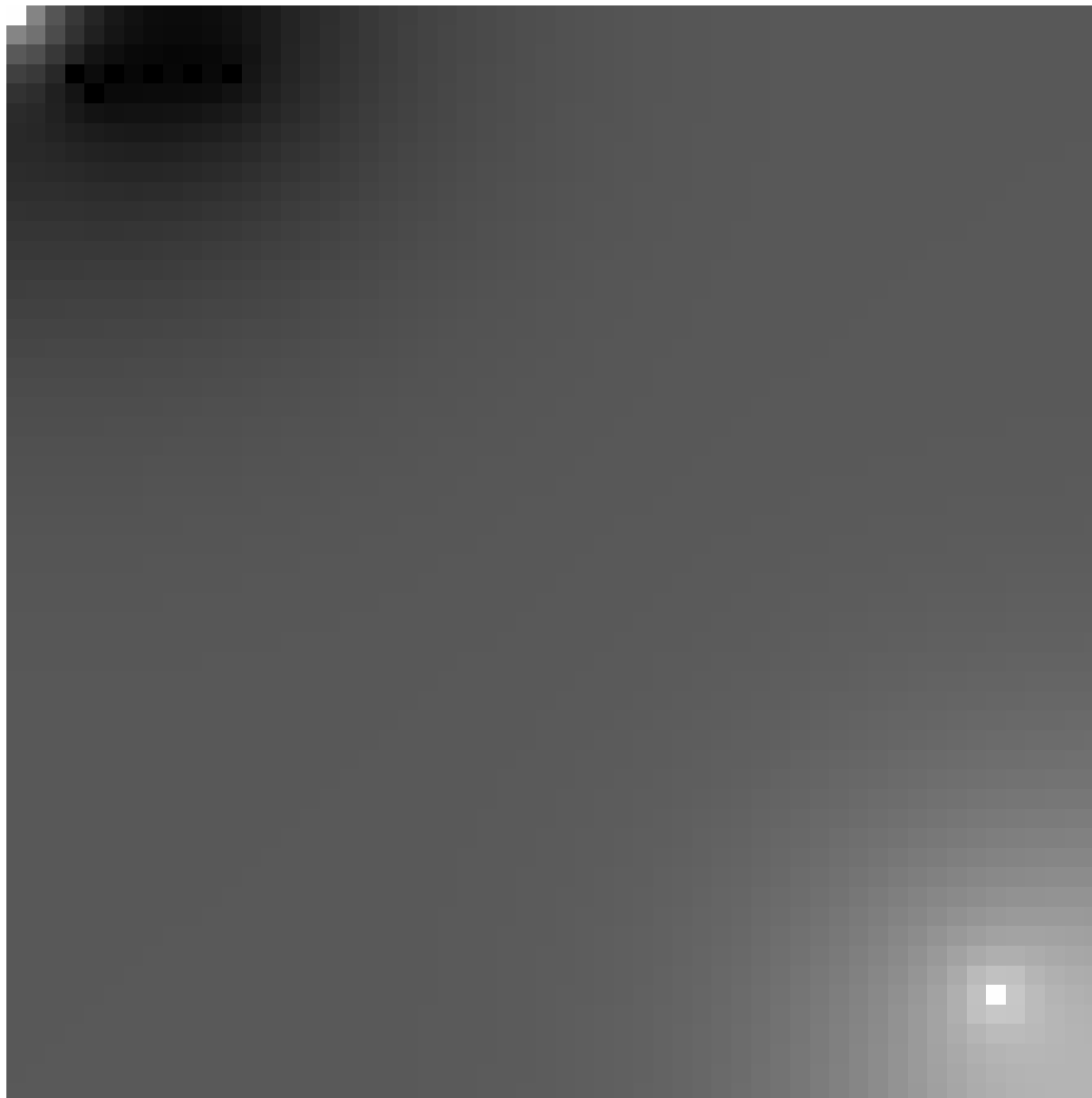
No wrap-around for edges

- **Spots with permanent temperature** (input)
- Iterative algorithm: repeat the computation until the difference between two consecutive iterations is negligible

$$\forall i, j: |t_{old}[i][j] - t_{new}[i][j]| \leq 0.00001$$

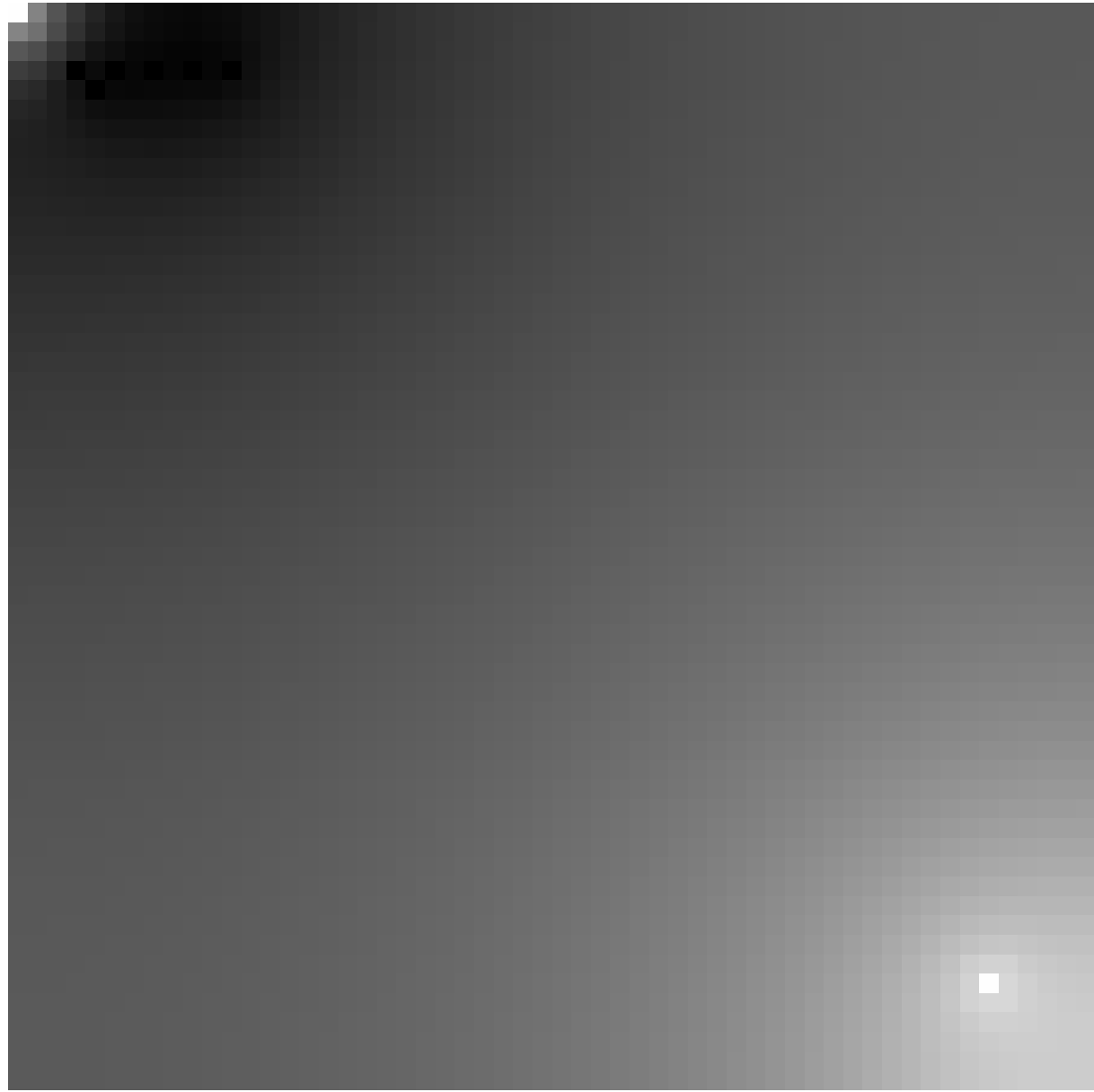


Example of evolution – 1



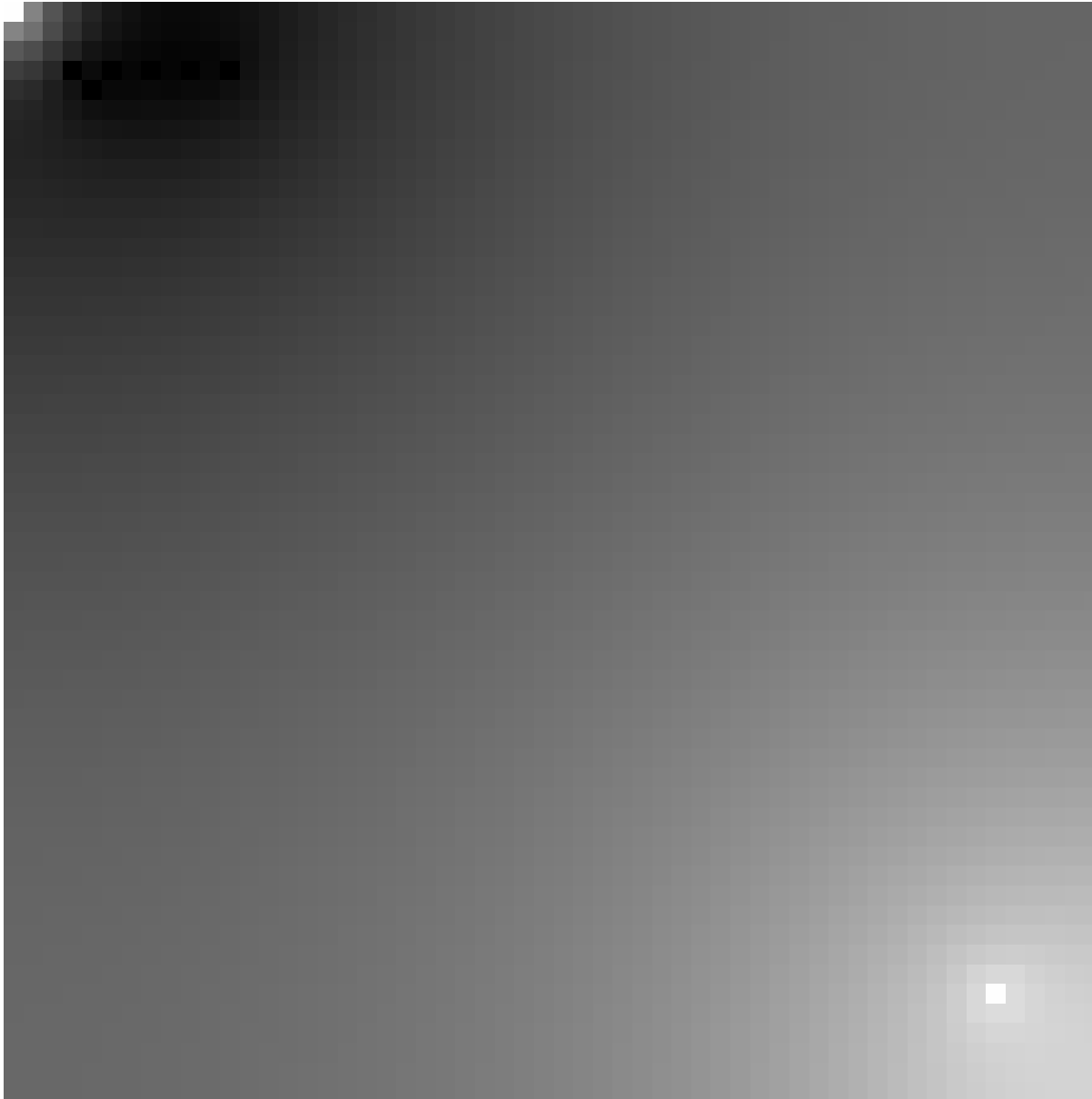


Example of evolution – 2



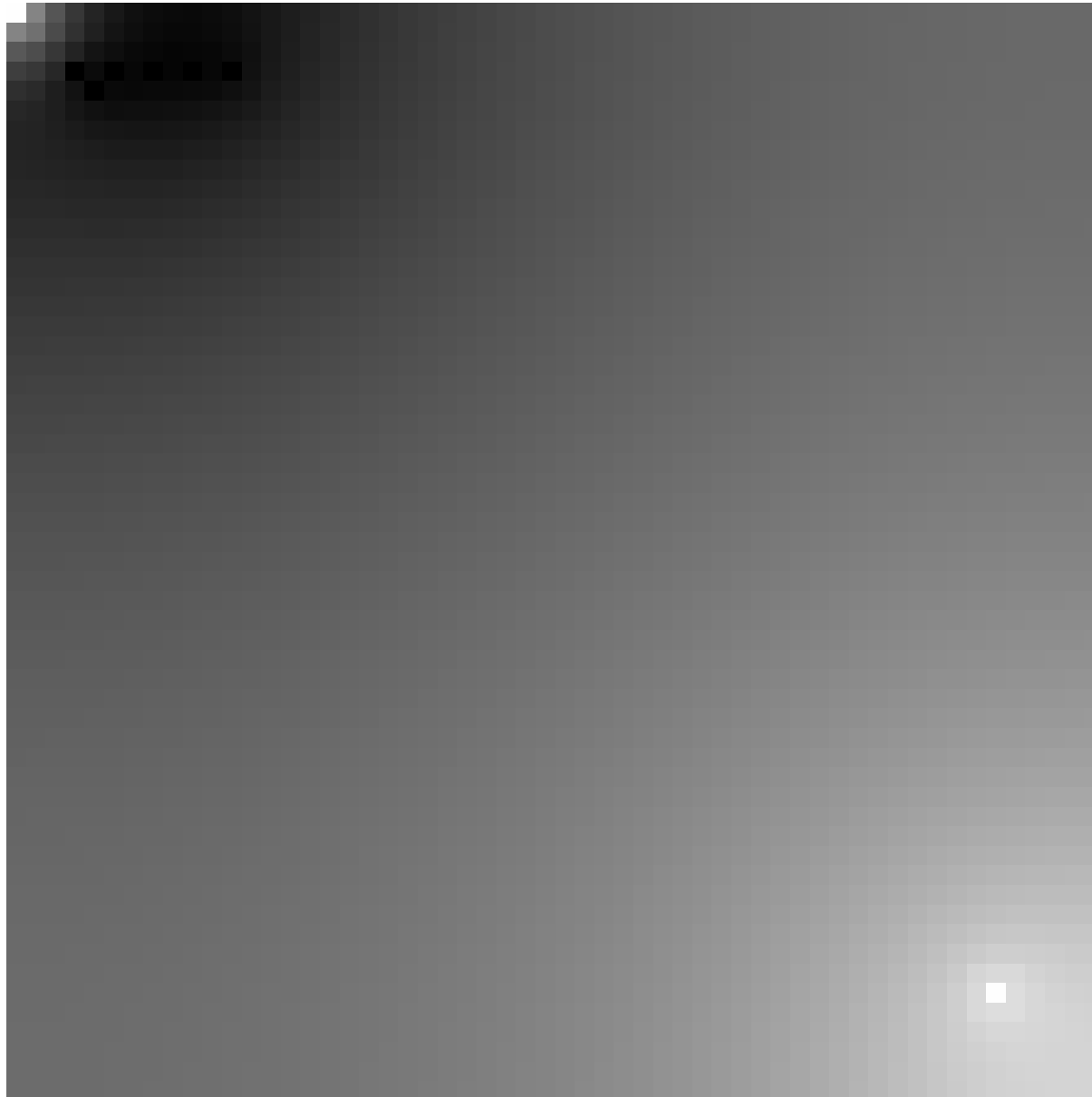


Example of evolution – 3



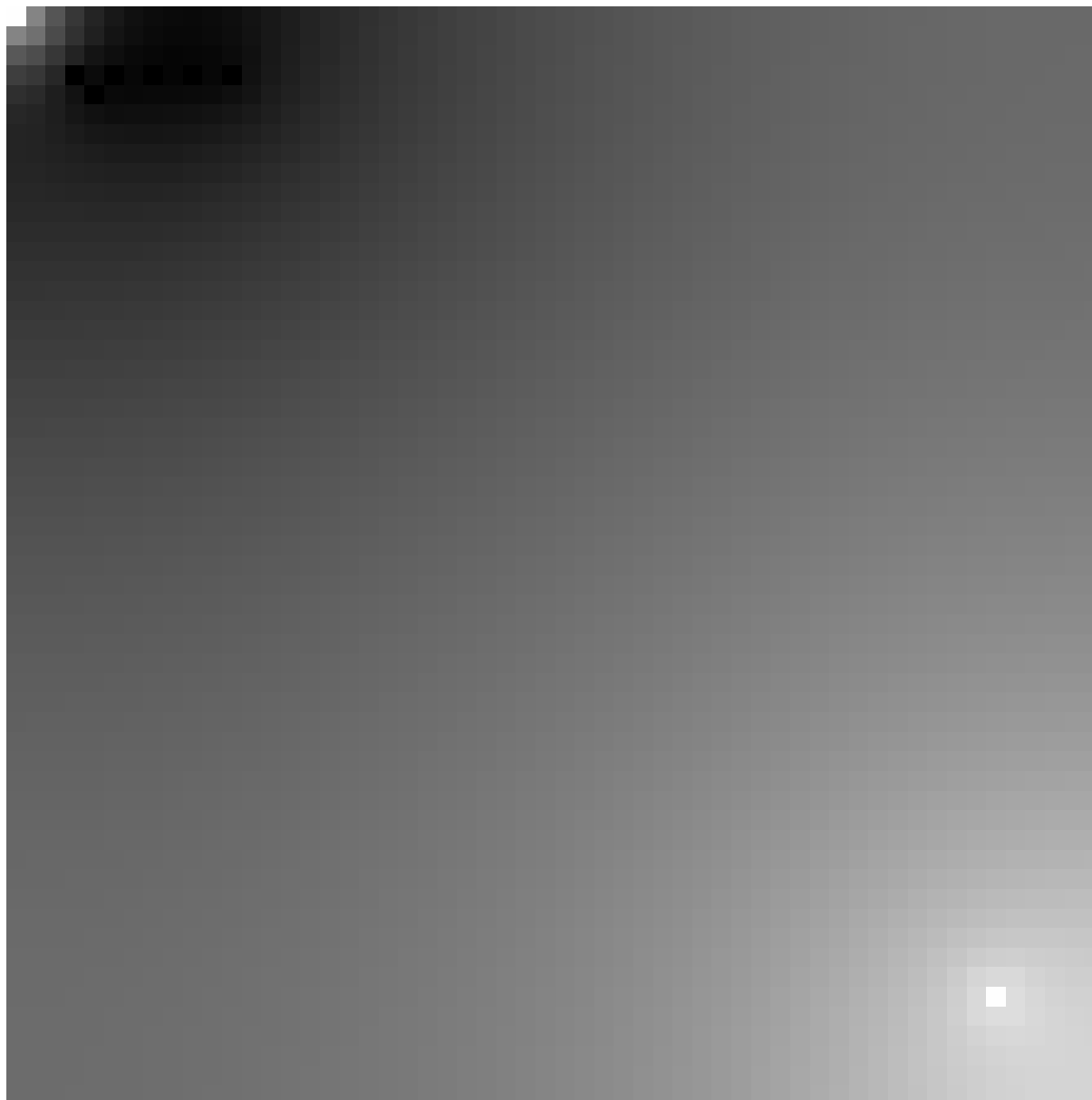


Example of evolution – 4





Example of evolution – 5





HW3 assignment

- Use the **code skeleton**
 - reads test problems, measures runtime
 - The program outputs an image in Netbpm format
- **Assignment:**
 - implement the Simple 2D Heat Diffusion simulator
 - use MPI
 - upload your solution to UploadSystem
- **Flags for g++ (used by UploadSystem)**
 - `-Ofast -std=c++14 -march=native`



Tricky issues

- Think about the partition of the input matrix among processes.
- Use floats for temperature computation.
- Initialize the cells having non-permanent temperature with 128