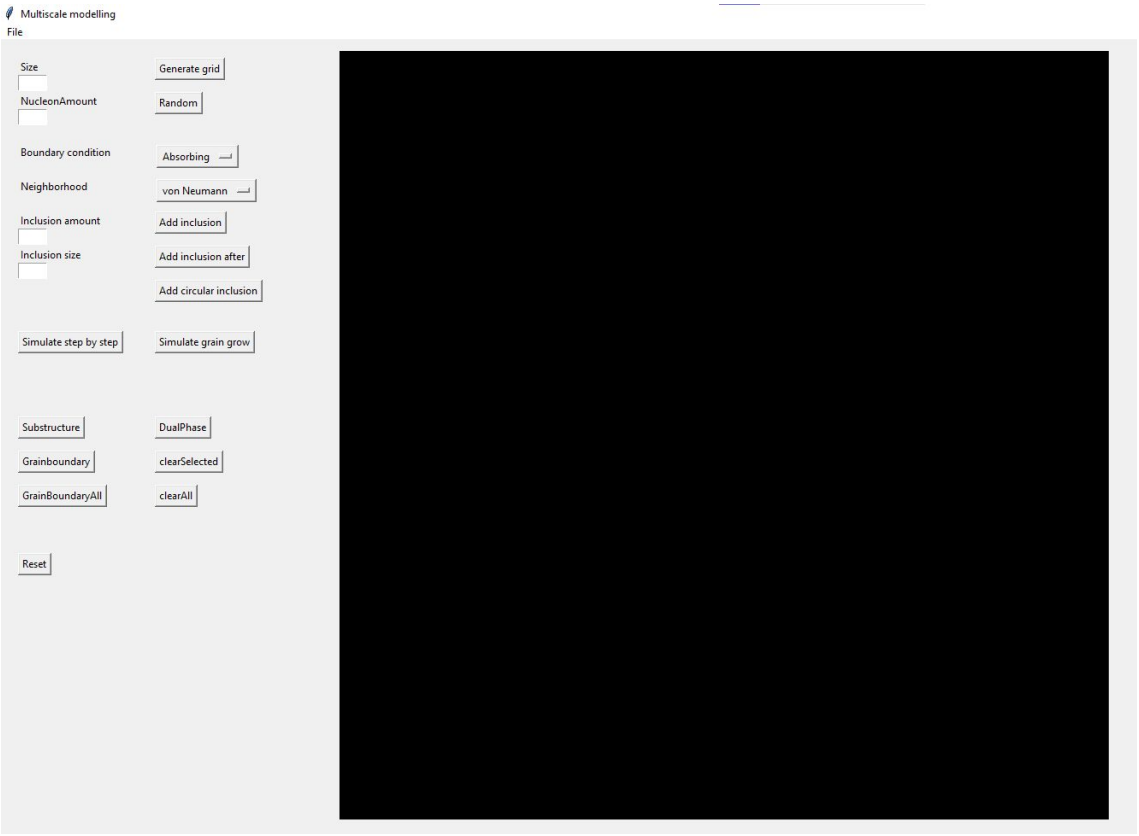
Jakub Łazarski - Multiscale modelling project report.

Report on the implementation of the program aimed at solving grain growth algorithm modifications.

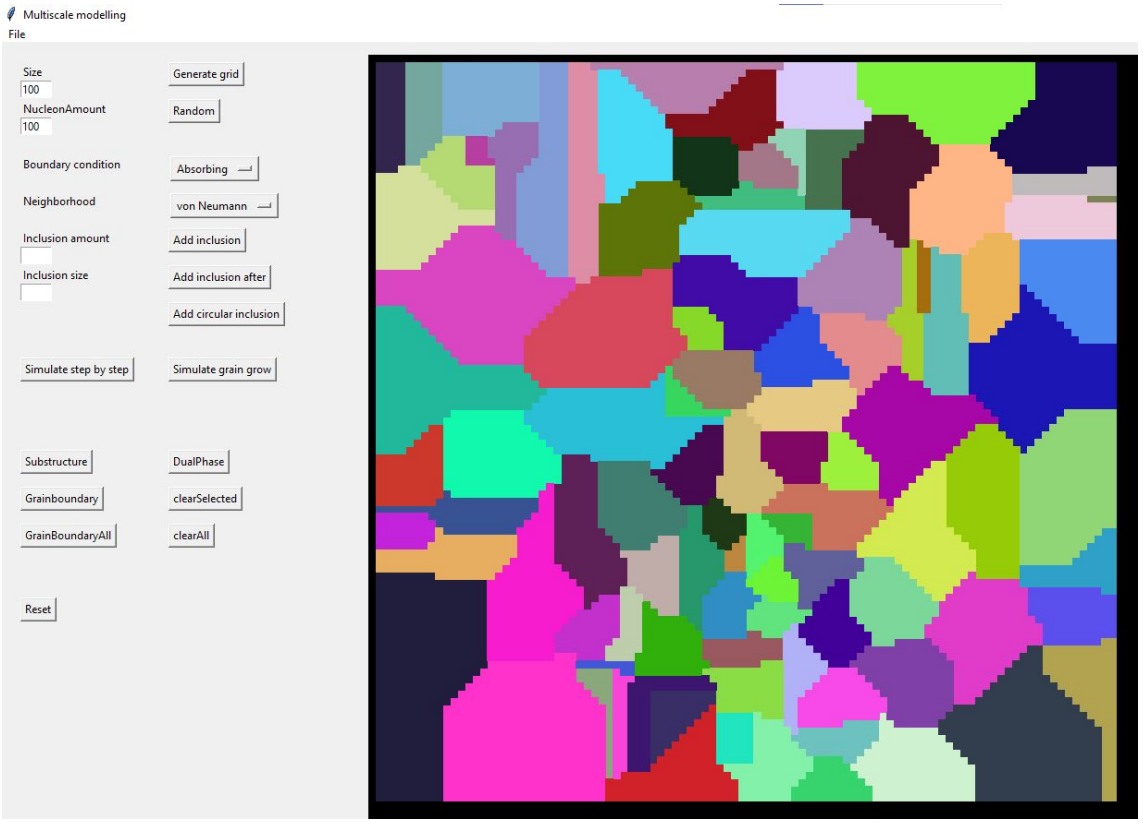
The program was built in according to the descriptions posted on the website. Nucleation has been completed with all neighbourhoods types and both boundary conditions.

Description of the individual buttons and their functionality are described below, along with screenshots corresponding to the action.

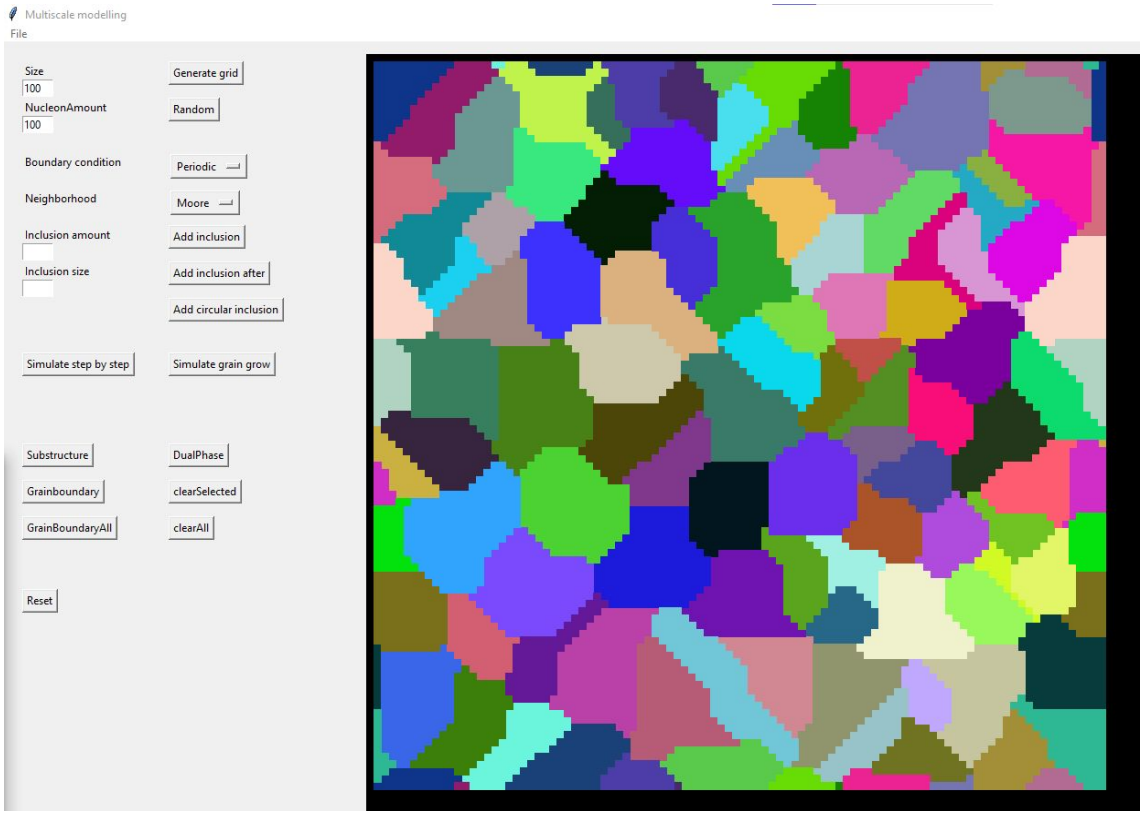
First screenshot shows startup screen:



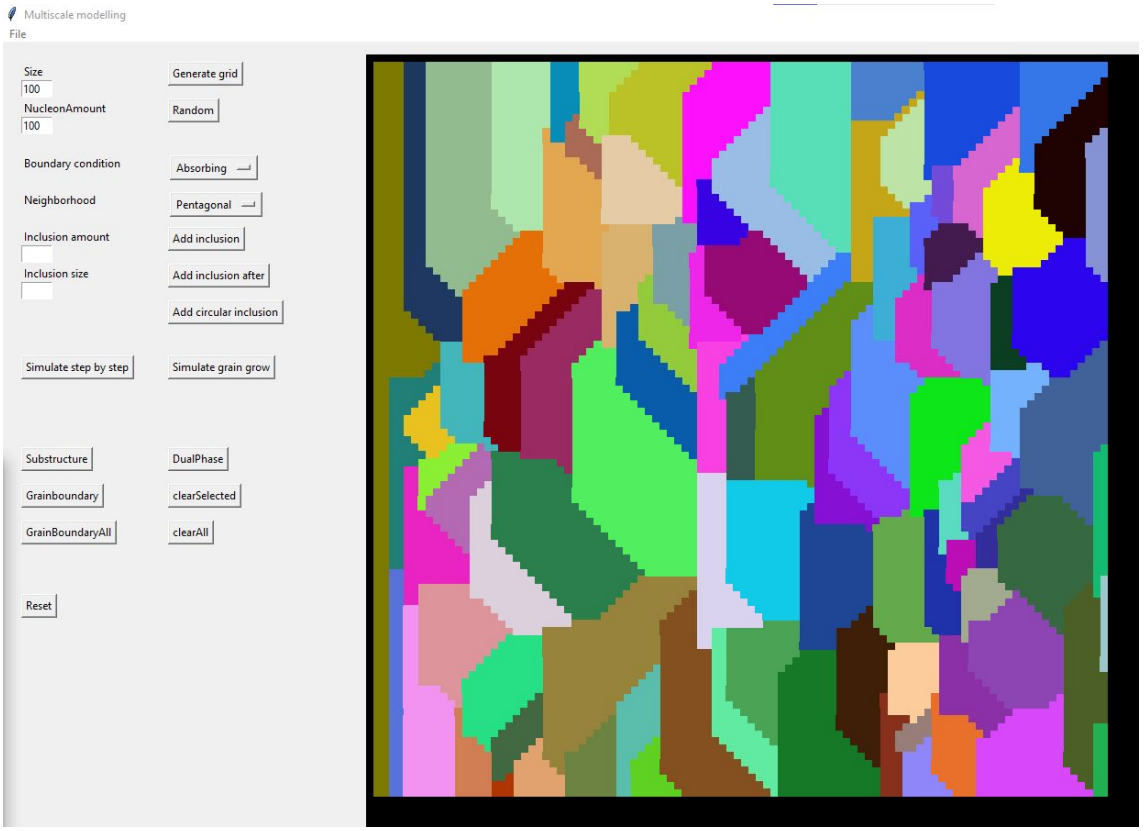
- Button “Generate grid” - generates a mesh of the sizes given in field "Size", which is used during nucleation
- Button “ Random” - randomly places seed embryos, the number of embryos is given in the field “NucleonAmount”
- Field “BoundaryCondition” - under the drop-down list there are two boundary conditions: "Absorbing" and "Periodic"
- Field “Neighborhood” - under the drop-down list there are neighbourhoods of the type “von Neumann”. “Moore”, “Pentagonal”, “Hexagonal” and“ ShapeControl”
- Button “Add inclusion” - allows you to add inclusions in the amount specified in the field “Inclusion amount” with the size specified in the “Inclusion size” field
- Button “Add inclusion after” - allows you to add inclusions after grain growth in the amount given in the field "Inclusion amount" with the size given in the field "Inclusion size"
- Button “Add circular inclusion” - allows you to add a circular inclusion amount in the field "Inclusion amount" with the radius specified in the field "Inclusion size"
- Button - “Simulate step by step” - enables grains to grow step by step
- Button “ Simulate grain growth” - enables the simulation of grain growth until the entire mesh is full
- Button - “Substructure” - allows you to create a substructure for selected grains
- Button “DualPhase” - makes it possible to add dual-phase grain growth
- Button “GrainBoundary” - allows you to create a border for selected grains, and the "clearSelected" button - clears the space
- Button “GrainBoundaryAll” - allows you to create a border for all grains, and the "clearAll" button - clears the space
- Button “Reset” - resets space and allows the growth process to be carried out again for new conditions
- Menu “File” - allows you to import and export space to a ".txt" file and close the program via the "Exit" button



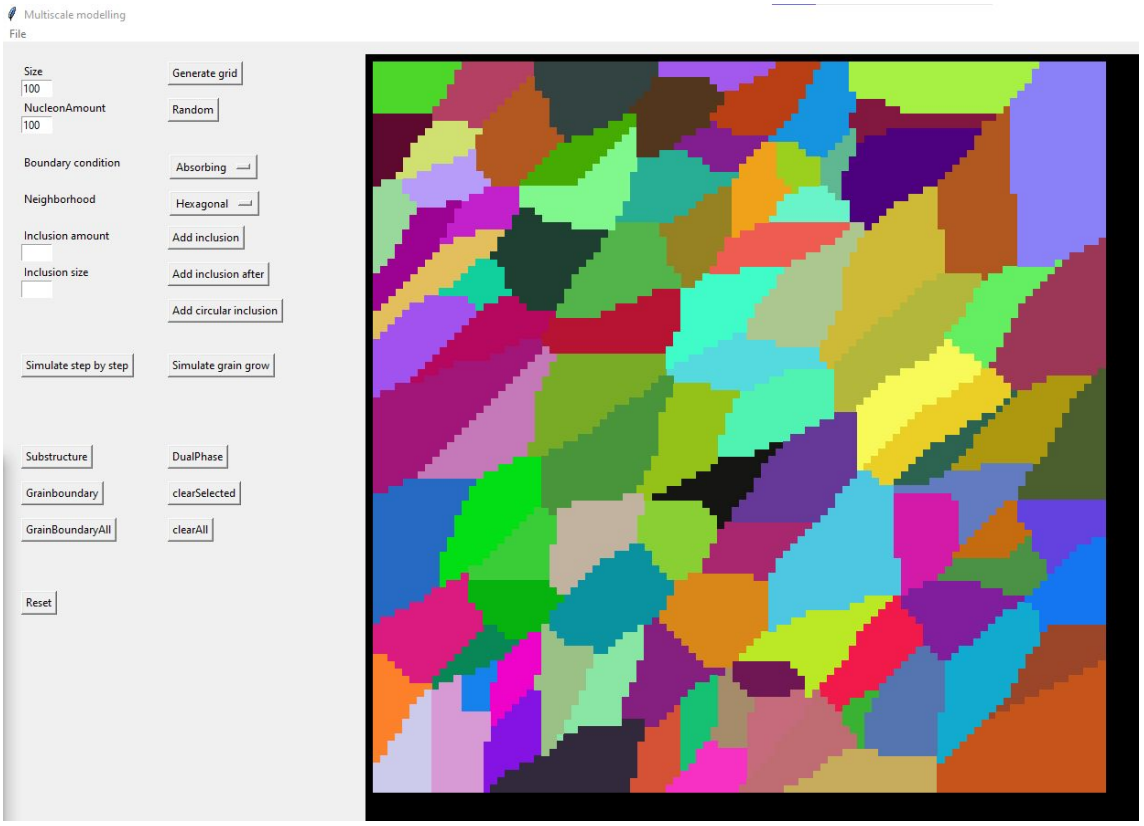
Grid 100x100
Nucleon amount 100
Boundary condition - Absorbing
Neighborhood - von Neumann



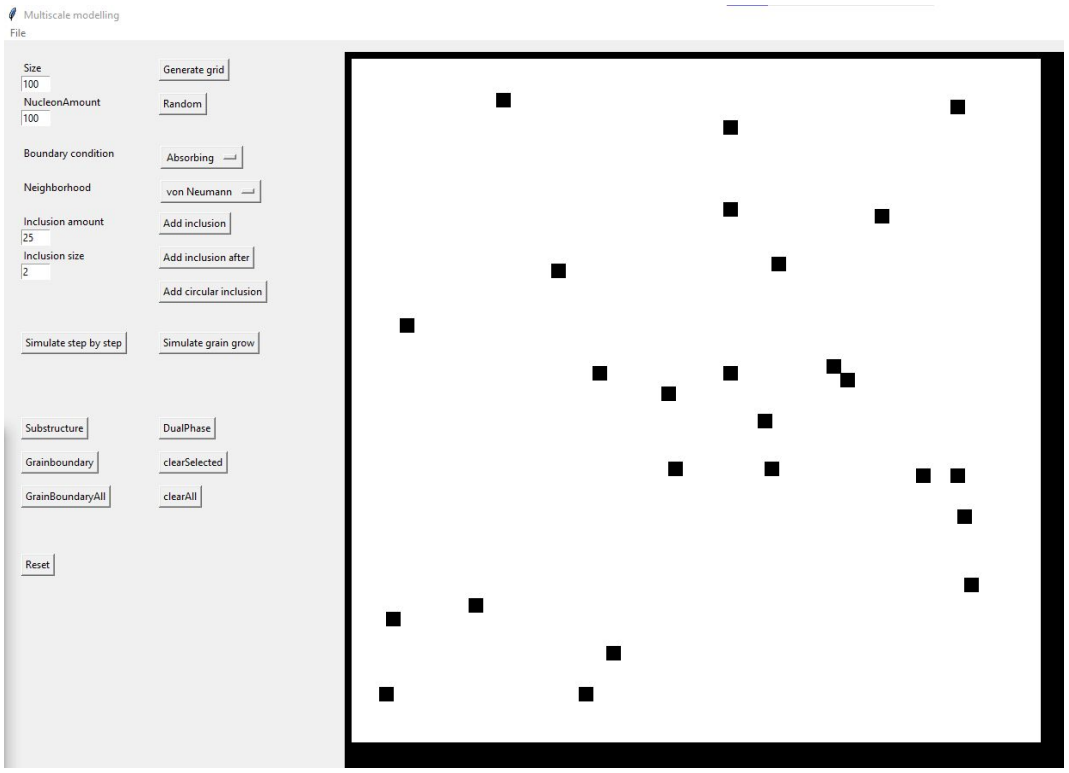
Grid 100x100
Nucleon amount 100
Boundary condition - periodic
Neighborhood - Moore



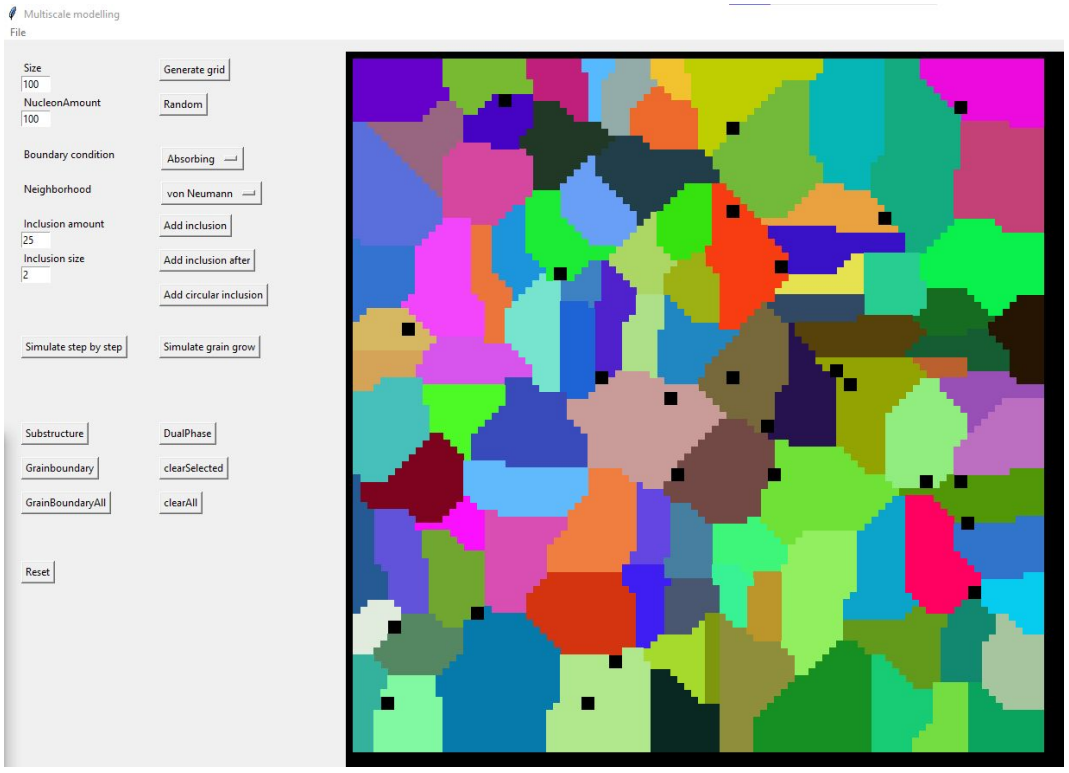
Grid 100x100
Nucleon amount 100
Boundary condition - Absorbing
Neighborhood - pentagonal

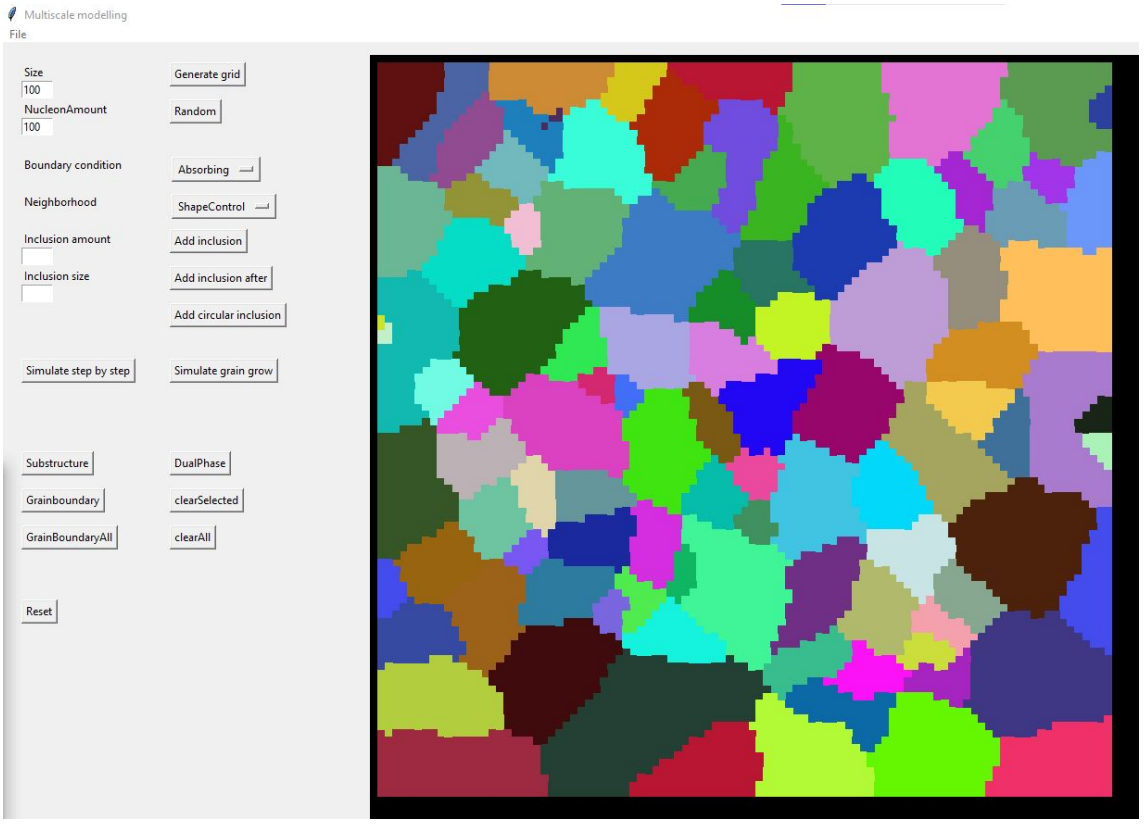


Grid 100x100
Nucleon amount 100
Boundary condition - Absorbing
Neighborhood - hexagonal

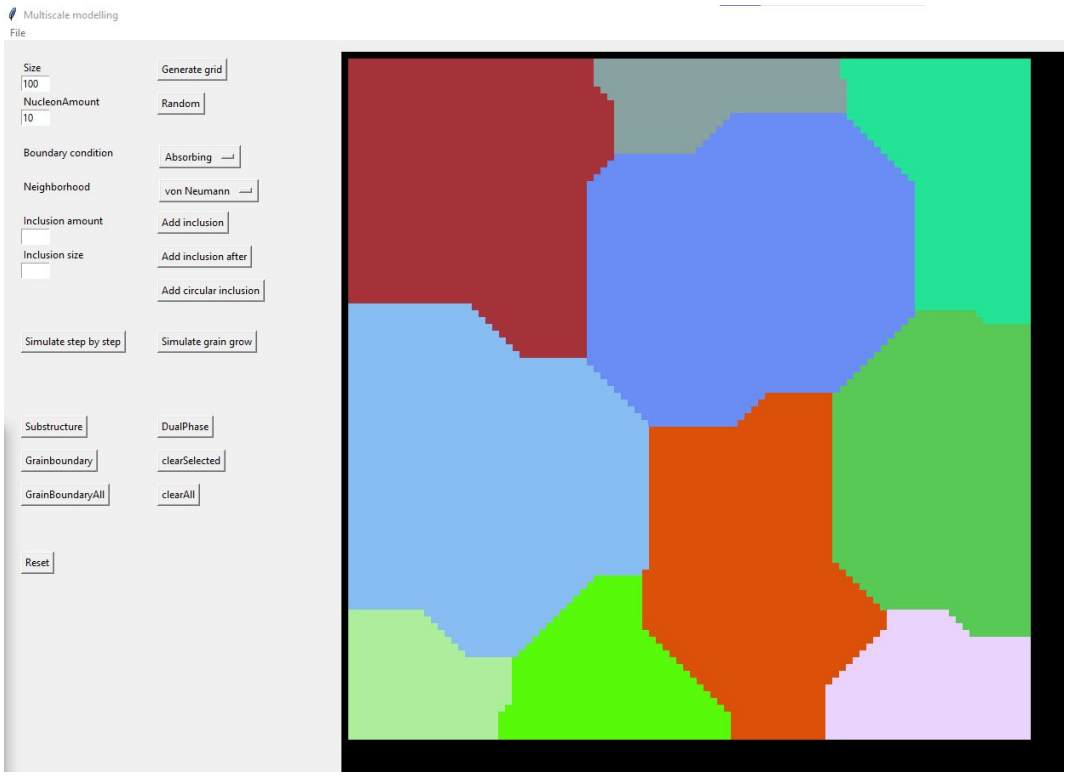


Grid 100x100
Nucleon amount 25
Boundary condition - Absorbing
Neighborhood - von Neumann
Inclusion size 2 - below screenshot after simulation

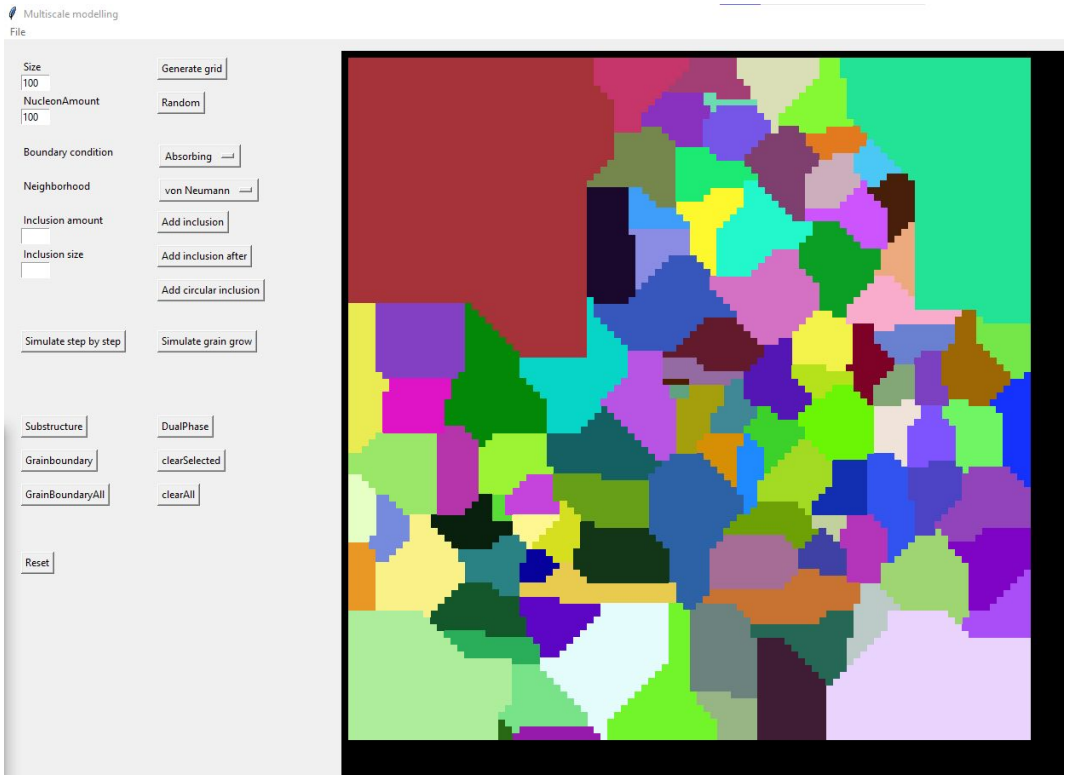


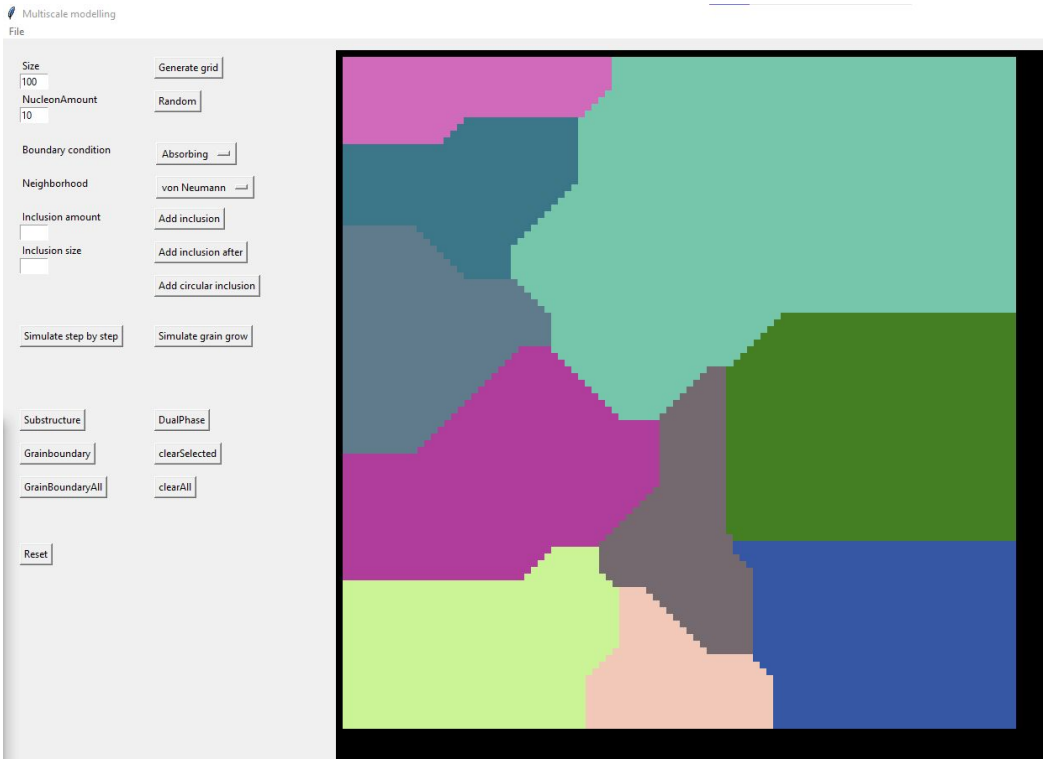


Grid 100x100
Nucleon amount 100
Boundary condition - Absorbing
Neighborhood - shape control with 10% probability for rule 4

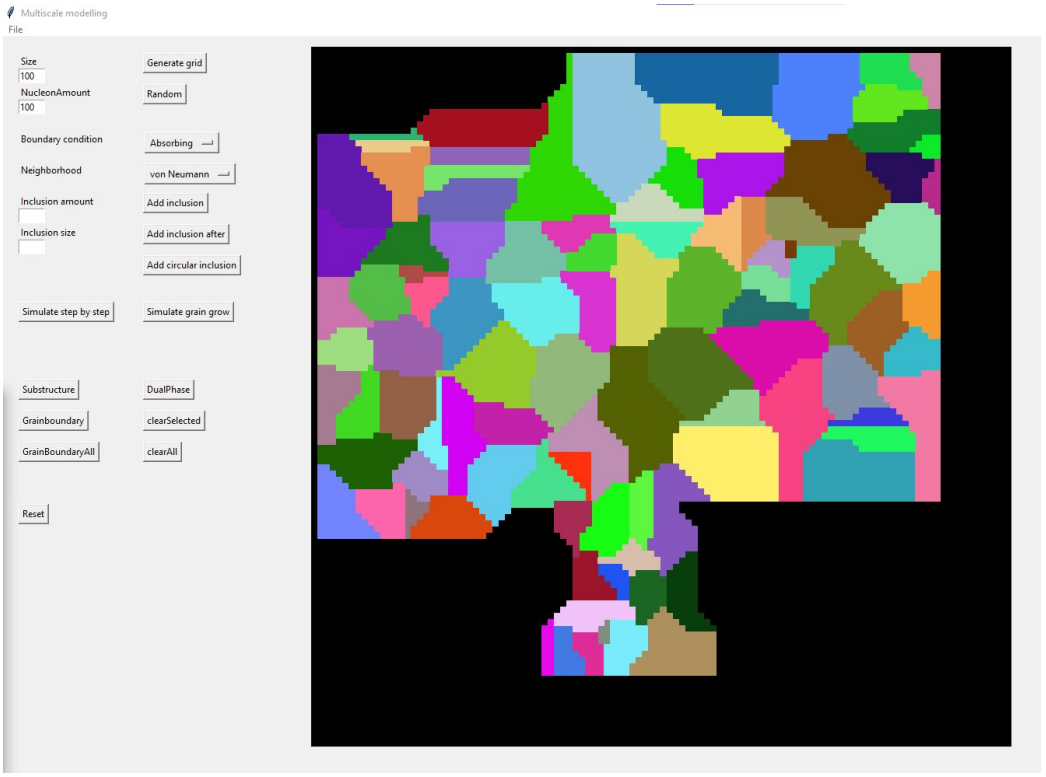


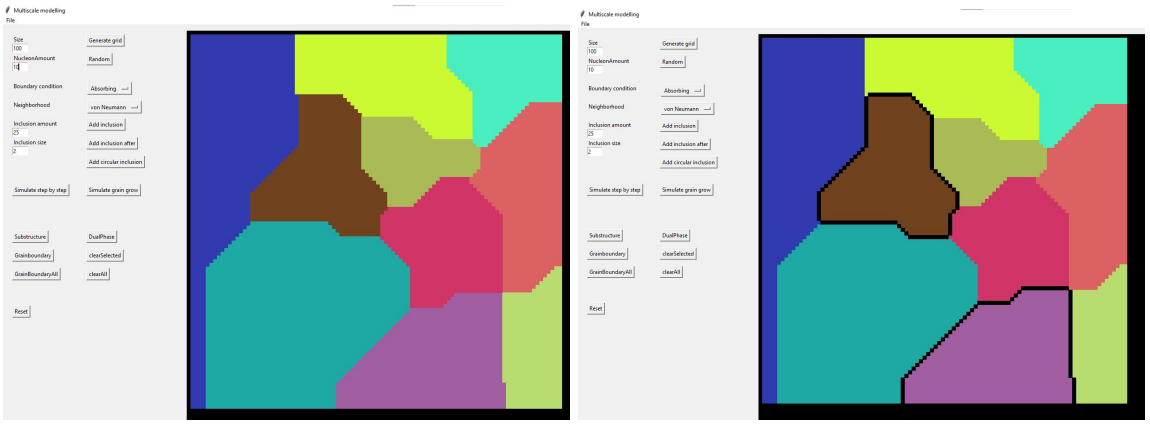
Grid 100x100
Nucleon amount 10
Boundary condition - Absorbing
Neighborhood - von Neumann
Substructure in all 4 corners - below screenshot after simulation



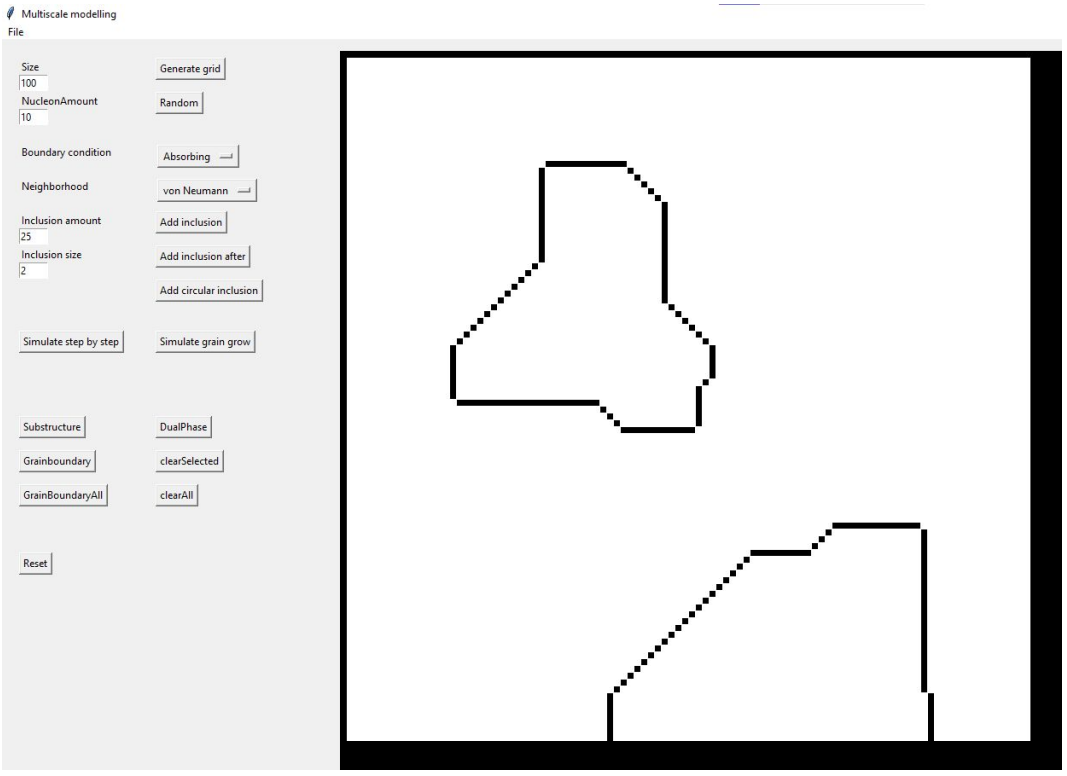


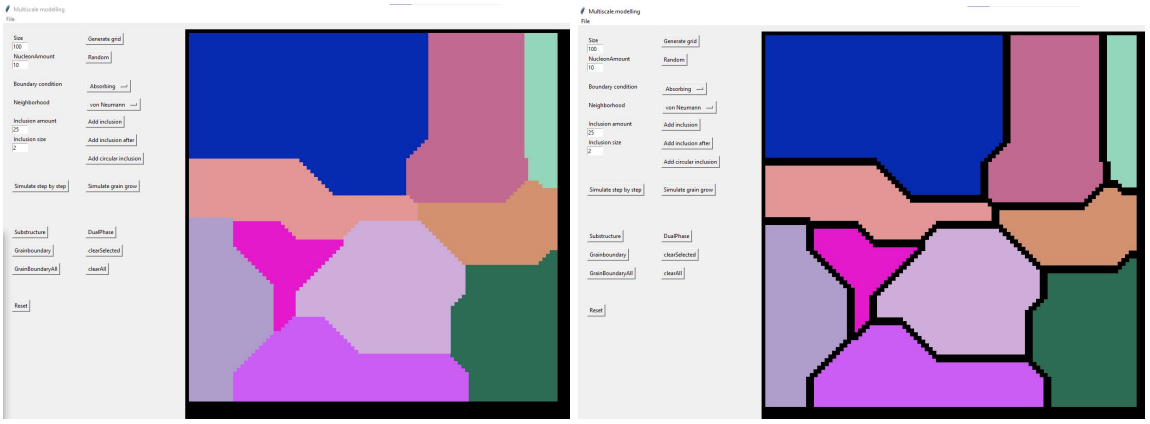
Grid 100x100
Nucleon amount 10
Boundary condition - Absorbing
Neighborhood - von Neumann
Dual phase in 3 corners - below screenshot after simulation



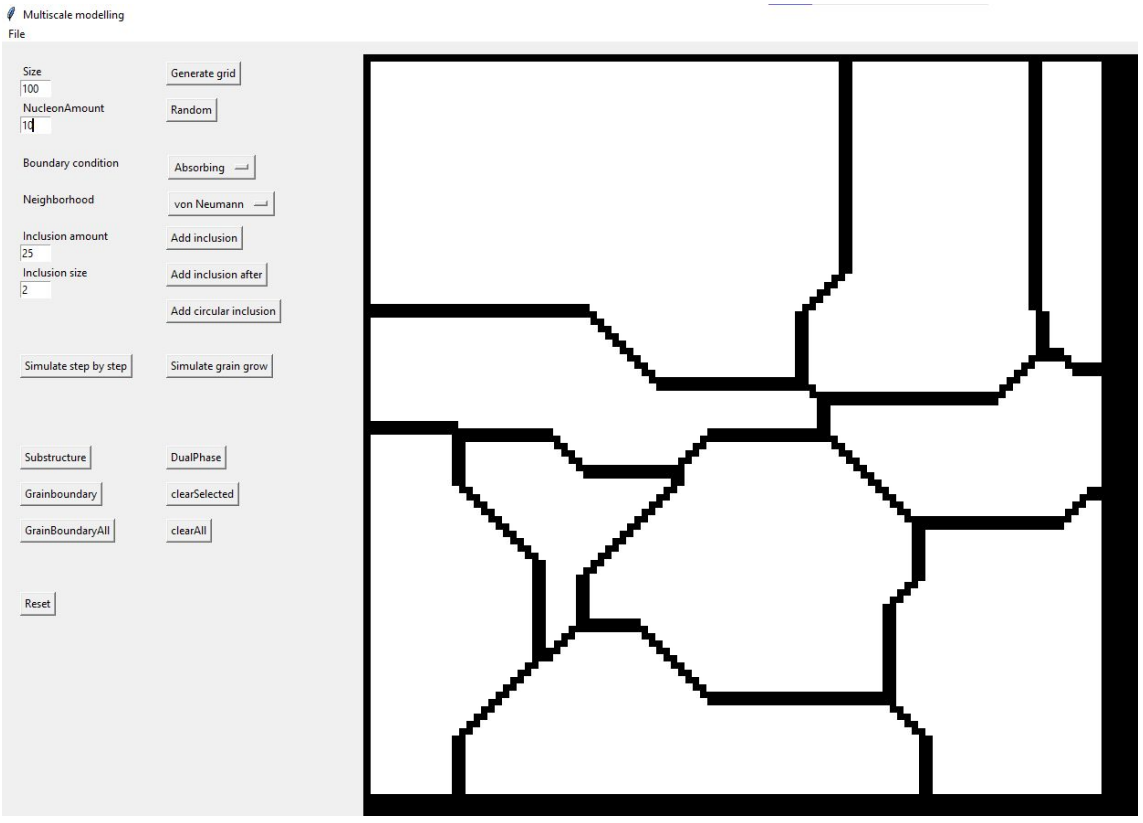


Grid 100x100
Nucleon amount 10
Boundary condition - Absorbing
Neighborhood - von Neumann
Grain boundary for selected grains - below screenshot after simulation





Grid 100x100
Nucleon amount 10
Boundary condition - Absorbing
Neighborhood - von Neumann
Grain boundary for all grains - below screenshot after simulation



Summary

To solve the problem of grain growth I used python wit three additional libraries:

- tkinter
- numpy
- random