Homework #1 (due Friday, July 7)

1 Synopsis

Write a program to simulate the growth of a two-dimensional interface as follows. On a square *grid* with a selectable size (e.g., 100x100), introduce *walkers* (or *particles*) according to one of the *walker generator models* detailed below. Move each walker on the grid according to one of the *walk models* until the walker is captured or leaves the grid. Run the simulation using different model settings and examine the final distribution of captured walkers graphically.

Please submit the solution to this homework *typeset* in the form of a report before the start of class on the due date. In addition, submit the software you write to Sasha via email by 6:00PM on the due date along with a brief note indicating how to run the code.

You are free to implement your simulator in any of C, C++, Java or Matlab and to produce graphics in any way you choose.

2 Details

You should design your simulator to execute a variety of different simulations involving different models for the motion of the walker, its capture, the boundary conditions, and how the walker is introduced to the grid.

- **Walk models**: At each iteration, have the walker move left or right according to one of the models listed below. After each lateral move, have the walker *descend* one grid position.
  - *No walk*: The walker does not move left or right. It simply falls.
  - *Symmetric random walk*: At each iteration, the walker moves left or right with equal probability. Use a random number generator to decide the move.
- **Capture models**: After each move (lateral or vertical), check if the walker should be captured according to one of the following models.
  - *No interaction*: The walker does not interact with previously captured walkers. It is captured (or removed) only by interaction with the bottom or lateral boundaries. This capture model is useful for testing.
  - *Random deposition*: The walker is captured if it contacts a previously captured walker *below it*, or if it touches bottom.
  - *Ballistic deposition*: The walker is captured if it contacts a previously captured walker *on any side* or if it touches bottom.

- **Lateral boundary conditions**:
  - *Absorbing*: If the walker crosses the left or right boundary, it is absorbed, i.e., disappears.
  - *Cyclic*: If the walker crosses the left or right boundary, it reappears on the opposite side.
- **Walker generators**: Introduce walkers at some fixed height just below the top of the grid. To avoid difficulties with the top boundary when executing with the *ballistic deposition* capture model, I suggest you introduce walkers one or two positions from the top.
  - *Center*: Introduce all walkers at the left-to-right midpoint of the grid.
  - *Uniform*: Introduce walkers according to a uniform distribution across the width of the grid.

3 Report

You will be evaluated on the *clarity* and *conciseness* of your presentation, as well as the quality of your analyses and your code. This exercise is intended to mimic a 'real' work assignment. It will be graded with that standard in mind.

Your report should include the following sections.

- **Summary**: What did you do? How did you do it? How did you test it? What results did you obtain?
- **Implementation**: What aspects of the problem did you model? How? What ambiguities did you encounter and how did you deal with them? What tools did you use?
- **Validation**: Describe what analysis you performed and what simulations you ran to validate your code. At a minimum, you should consider at least the following cases.
  - no walk, random deposition, absorbing, uniform
  - symmetric random walk, random deposition, absorbing, center

  Explain what aspects of your code these special cases test and what result(s) you expected and obtained. Include graphics and introduce subsections as needed.
- **Results**: Please describe the results you obtain, including graphics, for at least the following simulations.
  - no walk, ballistic deposition, absorbing, uniform
  - symmetric random walk, random deposition, absorbing, uniform
  - symmetric random walk, ballistic deposition, absorbing, center
  - symmetric random walk, ballistic deposition, absorbing, uniform

  If you have time, report additional results, for example, simulations employing cyclic boundary conditions.