Coding Assignment

Q[1] In olden days, a TV game show, popularly known as Let's Make a Deal (hosted by Monty Hall), the game is played as follows. There is a contestant and a TV host (like our KBC). Initially 3 closed doors are shown. The contestant picks up one of the three doors. In fact, there is a car behind one of the three and a goat each behind the other two doors. After the contestant picks up the door, the host opens one of the doors which has a goat (because, there are 2 goats, the host can open a door that hides a goat, no matter which door the contestant picks up!). The host then gives the contestant the option of abandoning the picked up door in favor of the unpicked and unopened door. The puzzle now is to determine which of the following 3 strategies, is the best suited for the contestant

(a) Always stay with the door initially picked
(b) Randomly stay (or) Switch (This can even be decided by a coin flip)
(c) Always switch to the unpicked and unopened door.

Q[2] A very simple minded simulation of stock price (which is a Brownian motion) involves a 1-D motion of a random walker as a function of time. On each day you are constrained to take only one of the following set S: {-2, -1, 0, +1, +2}, all of which are equally probable. Produce 10 such distinct trajectories each 10 years long and plot them. Build user defined Moving averages and plot them along with the stochastic movement.

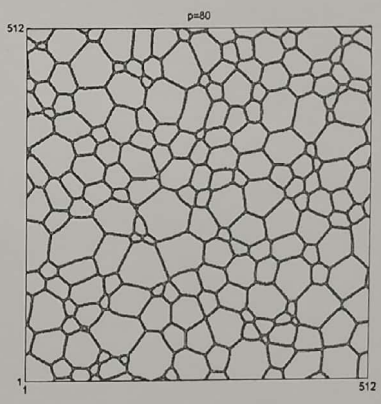
Q[3] Plan, design, develop and Simulate the game of life

Game of life is invented by John Conway, which models the process of birth, survival and death. The idea is that, organisms require others in order to survive and procreate, but overcrowding results in death. You can choose a 2-d array of size **NxN**. Each cell can be thought of either holding life (or) no-life form. The presence of life can be easily represented by a * symbol and a blank for the absence of the life form. The game starts with initial generation, which consists of a mix of * and blanks. The following **3** rules govern the transition from one generation to the next:

(a) **Birth Rule:** An organism is born into an empty cell that has exactly 3 living neighbors.
(b) **Survival Rule:** An organism with either 2 or 3 living neighbors survives from one generation to the next.
(c) **Death Rule:** An organism with 4 or more neighbors dies due to overcrowding. An organism with fewer than two neighbors dies from loneliness.

Q[4] The diagram depicts grains of different sizes on a 512x512 square grid (the data file posted and the plot which appears above could be different). The plot is generated on the basis of a given variable $\psi(x,y)$ at each grid point (the actual data file psi.dat containing the $\psi(x,y)$values are already posted... Answer the following metallurgical question of interest.

(i) What is the mean grain size ?
(ii) What is the frequency distribution for the grains ?
(iii) Comment on the nature of distribution.



Write a brief report on the Plan, design and the development process you have carried out. Submit your source code (both soft and hard copies) along with your report. We plan to optimize this code at a later stage.

Q[5] Develop a product based on Monte Carlo Potts Models that can be used to study the grain growth evolution of Nanomaterials. Perform 2D and 3D simulation of grain growth in Nano Structured material.

Computational Challenges:

1. Lattice Simulation
2. Plan, Design and Develop a workable code for the Monte Carlo Potts Model
3. Developing efficient Data Structures that would result in memory savings.
4. Optimizing the code to achieve better Performance in terms of spatial and temporal speed-up.
5. Visualisation (Matlab (or) OPENGL (or) TECPLOT etc…)

The interaction energy of the lattice points interacting with its neighbours are calculated, using the Potts Model. The Hamiltonian of the interaction is given below.

$$E = -J\sum_{j=1}^{n}\left(\delta_{S_i}\delta_{S_j} - 1\right)$$

Where J is a positive constant specifying a measure of interaction of the evaluated i-th point with the neighbouring points.

S_j = Orientation at a random site i; S_j = Orientation of nearest neighbour of i

n = nearest neighbours (8 for 2-d and 26 for 3-d) δ = Krnoecker's Delta function

Monte Carlo Simulation Algorithm: *Metropolis Algorithm*

In the most general form, the algorithm of the MC methods is represented as

1. Generate Random Numbers (Several Random number)
2. Select a new possible state of the system from the list of possible states depending on the generated random number
3. Accept or reject the transition in the selected possible state depending on the generated random number.
4. Repeat steps (1)-(3).

Algorithmic steps : After choosing the matrix (either 2-D square or 3D cubic Lattice) and filling it up with random number, the simulation begins. The four major steps of the algorithms are:

a) Calculation of the free energy of a chosen element of the matrix (G_i) with its present crystallographic orientation (S_i) based on its neighbourhood.

b) Random choice of the new crystallographic orientation for that element (S_f)

c) New Calculation of the free energy of the same element (G_f), but with the new crystallographic orientation (S_f).

d) Comparison of the two values (G_f-G_i). The orientation that minimise the energy is chosen.

The above four steps are repeated several times in random positions of the matrix. The overall result is a microscopic simulation of the free energy decay in the system, which is a driving force for grain growth in nanomaterials. Suitable visualization software can be used for visualization

The End