

## Assignment 8



*Due: 30th April.*

### 1 Robots falling from a table

You are given a C++ code (*serial.cpp*) that simulates robots walking on a table. The direction in which they walk depends on where they are on the table, and they can fall off the edge. The program prints the number of robots left on the table after every time step.

- a) Understand the code.
- b) (5 points) Modify the code such that it stops when there are no more robots on the table, and it takes a command line argument which is the random seed; thus we can explore different initial conditions.
- c) (5 points) Write a script to be able to run four different initial conditions at the same time using *GNU Parallel*.
- d) (30 points) In contrast with the previous question, where you ran 4 different cases, here, you should focus on a *single run* of the robots application. You are to speed up the code using OpenMP, as well as any other parts that you think may benefit. Remember to always use *default(none)*.
- e) (5 points) Write a job script to perform a scaling analysis, i.e., one that runs the code for values of `OMP_NUM_THREADS` ranging from 1 to 4.
- f) (5 points) Make a plot of the scaling and estimate the serial fraction  $f$

$$f = \frac{T_s}{T_s + NT_1}$$

where  $T_s$  denotes the time spent in the serial part of the computation,  $N$  is the number of threads and  $T_1$  is time each thread takes to run. Thus  $NT_1 = T_p$  denotes the time spent in the parallelizable part of the computation.