

Programming Assignment #1

Multicore Systems

Game of Life

- ▶ John Conway's game of life

- ▶ Cellular automation

- ▶ Universe

- ▶ infinite 2D orthogonal grid of square cells
- ▶ Cell is one of two states : live or dead
- ▶ Each time step, cell lives or dies depending on 8 neighbor cells

- ▶ Rules

- ▶ Any live cell with $\#live_neighbors < 2$ or $\#live_neighbors > 3$ will die
- ▶ Any live cell with $2 \leq \#live_neighbors \leq 3$, will live
- ▶ Any dead cell with $\#live_neighbors = 3$ will come to life

3D Game of Life

- ▶ **Cubic cells in 3D universe**
 - ▶ One cell will have 26 (3^3-1) neighbors
- ▶ **Rules**
 - ▶ $N = \#$ live neighbors
 - ▶ If a live cell, with $N < D1$ or $N > D2$, will die
 - ▶ If a dead cell, with $L1 < N < L2$, will come to life
 - ▶ Otherwise, unchanged
- ▶ For example,
 - ▶ $D1 = 5, D2 = 20, L1 = 10, L2 = 15$

Programming

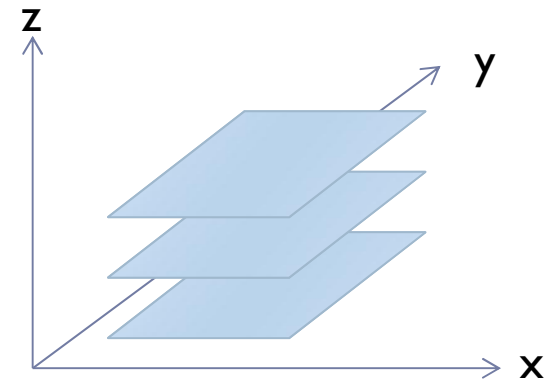
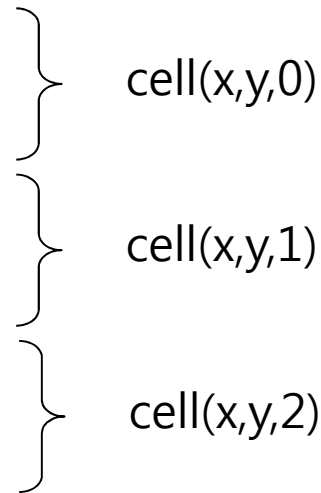
- ▶ 3D game of life in two programming models
 - ▶ Sequential programming
 - ▶ Parallel programming with Pthreads
- ▶ Input file
 - ▶ Size of universe, D1, D2, L1, L2, steps to run
 - ▶ Initial seeds of cells
- ▶ Output
 - ▶ Measured time
 - ▶ Report on monitor in microseconds (using `gettimeofday()` function)
 - ▶ Core work of simulation excluding file read/write, initialization
 - ▶ Output file
 - ▶ Resulting universe of cells (the same format of initial seeds of cells)

Input Example

▶ **Input**

- ▶ 3 x 3 x 3 universe input
- ▶ D1 = 5, D2 = 20, L1 = 10, L2 = 15
- ▶ 1000 steps

```
$ cat input.life
3 5 20 10 15 1000
010
010
101
110
100
000
001
001
101
```



Submission

- ▶ Submit through “icampus”
- ▶ Due by April 18, midnight (11:59:59 pm)
- ▶ studentID.tar
 - ▶ 2 directories for source code (sequential, pthread)
 - ▶ Each directory contains
 - ▶ Only source code of your work
 - NO binary/object code, input file, output file
 - ▶ Makefile (you should use optimization option `-O0`)
 - ▶ README : a short description on how to make and run

Additional items

- ▶ T.A. : Jungsik Choi (#85565)
 - ▶ jchoi@arcs.skku.edu
 - ▶ Will test the performance to measure your parallelization efforts
- ▶ Experiments environment
 - ▶ 16 Threads, 16GB RAM
- ▶ Penalty
 - ▶ Delayed submission: 10% per day, up to 50% of your score
 - ▶ 100% points will be deducted for a copied work

Teams

#	Name	#	Name
1	박종규	9	권초록
	고은석		조현규
2	김민규	10	이정진
	박주연		노윤호
3	이재욱	11	Rodrigo Melo
	정호주		Vaclavek Jonas
4	홍상기	12	김성민
	배종현		현병훈
5	박태순	13	김영대
	이성제		박성현
6	Markus Eriksson	14	김인규
	Ricky Cheung		김종원
7	정남훈	15	유길준
	최창인		최태림
8	손동식	Total 15 teams	
	이광현		