

Ιωάννης Κουτουλάκης 1115200900080  
Κωνσταντίνος Πατσουράκος 1115200900129  
Παράλληλα Συστήματα 2012-13  
Εργασία MPI (+ OpenMP): Το Παιχνίδι της Ζωής του John Conway

Έχουν υλοποιηθεί , MPI , MPI + OpenMP, Cuda.

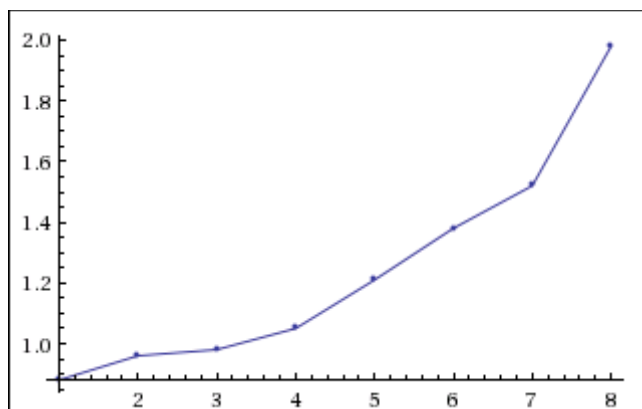
Παραδοχές που έχουν γίνει :

Αρχικά για MPI δεν δημιουργούμε το μεγάλο  $M \times N$  grid, απλά το σπάμε σε submatrices και κάνουμε εκεί μέσα rand το πρώτο generation.

MPI:

Η λογική που ακολουθήσαμε είναι: Σκεφτήκαμε να βρούμε το μέγεθος των υποπινάκων και να τους δημιουργούμε σε κάθε pc και το κάθε pc να φτιάχνει τον πίνακα με την χρήση της συνάρτησής rand(). Στη συνέχεια κάθε pc βρίσκει τους γείτονες χρησιμοποιώντας το rank του ( 0 η πρώτη διεργασία, 1 η δεύτερη...). Για παράδειγμα τον δεξιά γείτονα τον βρίσκουμε πηγαίνοντας στο  $(rank+1) \bmod N$  (σε  $N \times N$  πίνακα) για να βρούμε την στήλη και  $N/(rank \div N)$  για την γραμμή και το αποτέλεσμα mod το πλήθος των διεργασιών. Σε αυτό το σημείο, έχουμε φτιάξει την επικοινωνία μεταξύ των γειτόνων, τον πίνακα με τα random αρχικά δεδομένα(current\_gen), αυτόν που θα αποθηκεύσουμε μέσα την επόμενη γεννεά(next\_gen) μας +2 εξτρά στήλες και γραμμές στα άκρα τους για να βάλουμε την πληροφορία που θα πάρουμε από τους γείτονες, και αρχίζουμε να υπολογίζουμε την επόμενη γεννεά. Αρχικά στέλνουμε στους γείτονές μας τα στοιχεία που χρειάζονται ( ο πάνω γείτονας την πάνω γραμμή ο πάνω δεξιά το πάνω δεξιά στοιχείο του πίνακά μας κτλ) και λαμβάνουμε την αντίστοιχη πληροφορία που χρειάζεται για μας. Συνεχίζουμε με τον υπολογισμό της επόμενης γεννεάς των εσωτερικών στοιχείων που είναι ανεξάρτητα από τα στοιχεία που περιμένουμε από τους γείτονες, και έπειτα βάζουμε barrier στην διεργασία μας, να περιμένει τις άλλες να φθάσουν σε αυτό το σημείο (MPI\_Waitall) και κάνουμε swar τους 2 πίνακές μας ( current\_gen, next\_gen ). Έπειτα κάνουμε AllReduce (όταν δεν το θέλουμε στις μετρήσεις μας το βάζουμε σε σχόλια ) και έπειτα επαναλαμβάνουμε την διαδικασία για την επόμενη γεννεά.

Ενδεικτικές εκτελέσεις:

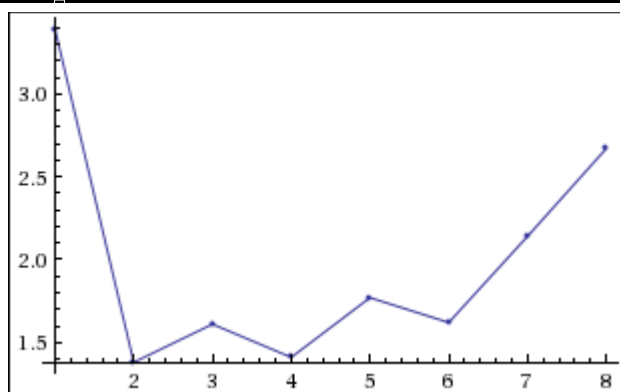


## 25 διεργασίες

```
linux03:/home/users/std09080/Desktop/para1_koutou>time mpiexec -np 25 ./a.out 10 840 && time mpiexec -np 25 ./a.out 20 840 && time mpiexec -np 25 ./a.out 40 840 && time mpiexec
-np 25 ./a.out 10 1680 && time mpiexec -np 25 ./a.out 20 1680 && time mpiexec -np 25 ./a.out 30 1680 && time mpiexec -np 25 ./a.out 10 3360 && time mpiexec -np 25 ./a.out 20 3
360 && time mpiexec -np 25 ./a.out 30 3360 && time mpiexec -np 25 ./a.out 10 6720 && time mpiexec -np 25 ./a.out 20 6720 && time mpiexec -np 25 ./a.out 40 6720 && time mpiexec
-np 25 ./a.out 10 13440
gens 10 dim 840
0.152u 0.008s 0:00.88 17.0%      0+0k 0+0io 0pf+0w
gens 20 dim 840 (timestamp)
0.136u 0.024s 0:00.96 15.6%      0+0k 0+0io 0pf+0w
gens 40 dim 840 (lock1)
0.132u 0.028s 0:00.96 15.6%      0+0k 0+0io 0pf+0w
gens 10 dim 1680 (append(timestamp t))
0.132u 0.028s 0:01.04 14.4%      0+0k 0+0io 0pf+0w
gens 20 dim 1680
0.136u 0.024s 0:01.21 12.3%      0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.136u 0.024s 0:01.38 10.8%      0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.144u 0.016s 0:01.52 9.8%       0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.148u 0.012s 0:01.98 7.5%       0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.156u 0.004s 0:02.56 5.8%       0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.140u 0.020s 0:03.36 4.7%       0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.152u 0.008s 0:05.34 2.8%       0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.148u 0.012s 0:09.47 1.5%       0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.144u 0.020s 0:10.71 1.4%       0+0k 0+0io 0pf+0w
```

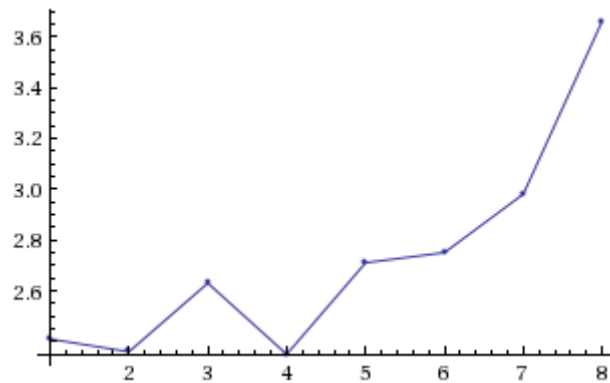
## 36 διεργασίες

```
linux03:/home/users/std09080/Desktop/para1_koutou>time mpiexec -np 36 ./a.out 10 840 && time mpiexec -np 36 ./a.out 20 840 && time mpiexec -np 36 ./a.out 40 840 && time mpiexec
-np 36 ./a.out 10 1680 && time mpiexec -np 36 ./a.out 20 1680 && time mpiexec -np 36 ./a.out 30 1680 && time mpiexec -np 36 ./a.out 10 3360 && time mpiexec -np 36 ./a.out 20 3
360 && time mpiexec -np 36 ./a.out 30 3360 && time mpiexec -np 36 ./a.out 10 6720 && time mpiexec -np 36 ./a.out 20 6720 && time mpiexec -np 36 ./a.out 40 6720 && time mpiexec
-np 36 ./a.out 10 13440
gens 10 dim 840
0.144u 0.036s 0:03.39 5.0%       0+0k 5184+0io 22pf+0w
gens 20 dim 840 (timestamp)
0.128u 0.036s 0:01.38 10.8%      0+0k 0+0io 0pf+0w
gens 40 dim 840 (lock1)
0.140u 0.024s 0:01.61 9.9%       0+0k 0+0io 0pf+0w
gens 10 dim 1680 (append(timestamp t))
0.120u 0.044s 0:01.41 11.3%      0+0k 0+0io 0pf+0w
gens 20 dim 1680
0.136u 0.028s 0:01.77 8.4%       0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.140u 0.020s 0:01.62 9.8%       0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.152u 0.012s 0:02.14 7.4%       0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.144u 0.016s 0:02.67 5.6%       0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.148u 0.016s 0:03.15 4.7%       0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.132u 0.032s 0:04.22 3.7%       0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.144u 0.016s 0:06.55 2.2%       0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.136u 0.028s 0:11.21 1.3%       0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.148u 0.016s 0:12.75 1.1%       0+0k 0+0io 0pf+0w
```



## 49 διεργασίες

```
linux03:/home/users/std09080/Desktop/para1_koutou>time mpiexec -np 49 ./a.out 10 840 && time mpiexec -np 49 ./a.out 20 840 && time mpiexec -np 49 ./a.out 40 840 && time mpiexec
-np 49 ./a.out 10 1680 && time mpiexec -np 49 ./a.out 20 1680 && time mpiexec -np 49 ./a.out 30 1680 && time mpiexec -np 49 ./a.out 10 3360 && time mpiexec -np 49 ./a.out 20 3
360 && time mpiexec -np 49 ./a.out 30 3360 && time mpiexec -np 49 ./a.out 10 6720 && time mpiexec -np 49 ./a.out 20 6720 && time mpiexec -np 49 ./a.out 40 6720 && time mpiexec
-np 49 ./a.out 10 13440
gens 10 dim 840
0.148u 0.016s 0:02.41 6.2%      0+0k 0+0io 0pf+0w
gens 20 dim 840 timestamp
0.160u 0.008s 0:02.36 6.7%      0+0k 0+0io 0pf+0w
gens 40 dim 840 lock();
0.148u 0.020s 0:02.63 6.0%      0+0k 0+0io 0pf+0w
gens 10 dim 1680 lapsedtime(timestamp, t);
0.136u 0.028s 0:02.35 6.3%      0+0k 0+0io 0pf+0w * 1000000;
gens 20 dim 1680
0.144u 0.020s 0:02.71 5.9%      0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.152u 0.012s 0:02.75 5.8%      0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.140u 0.024s 0:02.98 5.3%      0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.136u 0.028s 0:03.66 4.0%      0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.136u 0.032s 0:04.03 3.9%      0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.144u 0.020s 0:04.77 3.3%      0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.144u 0.020s 0:07.02 2.2%      0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.136u 0.032s 0:11.12 1.4%      0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.156u 0.012s 0:12.55 1.2%      0+0k 0+0io 0pf+0w
```

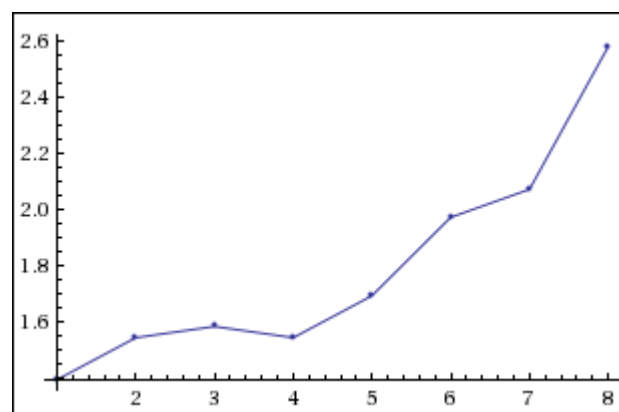


Με το Openmp άπλα κάναμε παράλληλα τον υπολογισμό του εσωτερικού πίνακα.

Ενδεικτικές εκτελέσεις:

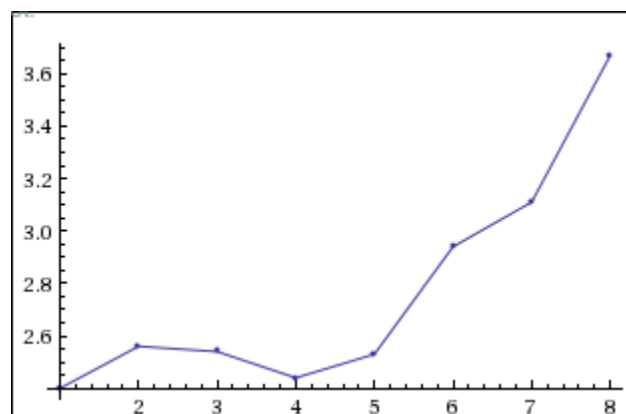
36 διεργασίες

```
linux03:/home/users/std09080/Desktop/paral_koutou>time mpiexec -np 36 ./a.out 10 840 && \
? time mpiexec -np 36 ./a.out 20 840 && \
? time mpiexec -np 36 ./a.out 40 840 && \
? time mpiexec -np 36 ./a.out 10 1680 && \
? time mpiexec -np 36 ./a.out 20 1680 && \
? time mpiexec -np 36 ./a.out 30 1680 && \
? time mpiexec -np 36 ./a.out 10 3360 && \
? time mpiexec -np 36 ./a.out 20 3360 && \
? time mpiexec -np 36 ./a.out 30 3360 && \
? time mpiexec -np 36 ./a.out 10 6720 && \
? time mpiexec -np 36 ./a.out 20 6720 && \
? time mpiexec -np 36 ./a.out 40 6720 && \
? time mpiexec -np 36 ./a.out 10 13440
gens 10 dim 840
0.140u 0.020s 0:01.39 11.5% 0+0k 0+0io 0pf+0w
gens 20 dim 840
0.140u 0.020s 0:01.54 10.3% 0+0k 0+0io 0pf+0w
gens 40 dim 840
0.140u 0.020s 0:01.58 10.1% 0+0k 0+0io 0pf+0w
gens 10 dim 1680
0.136u 0.024s 0:01.54 9.7% 0+0k 0+0io 0pf+0w
gens 20 dim 1680
0.148u 0.012s 0:01.69 8.8% 0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.156u 0.004s 0:01.97 7.6% 0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.152u 0.008s 0:02.07 7.2% 0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.152u 0.008s 0:02.58 5.8% 0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.140u 0.020s 0:03.27 4.8% 0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.140u 0.020s 0:04.25 3.7% 0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.136u 0.024s 0:06.54 2.2% 0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.144u 0.020s 0:11.13 1.4% 0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.148u 0.012s 0:12.67 1.1% 0+0k 0+0io 0pf+0w
```



49 διεργασίες:

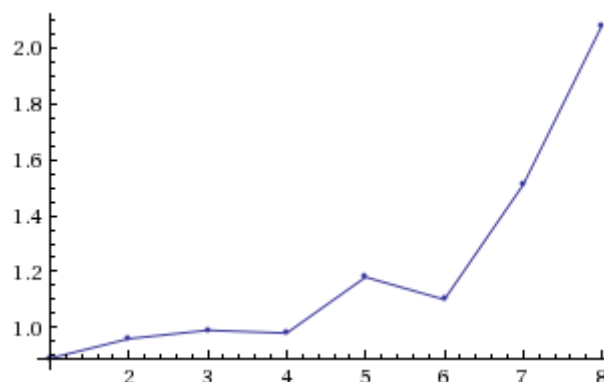
```
linux03:/home/users/std09080/Desktop/paral_koutou>time mpiexec -np 49 ./a.out 10 840 && \
? time mpiexec -np 49 ./a.out 20 840 && \
? time mpiexec -np 49 ./a.out 40 840 && \
? time mpiexec -np 49 ./a.out 10 1680 && \
? time mpiexec -np 49 ./a.out 20 1680 && \
? time mpiexec -np 49 ./a.out 30 1680 && \
? time mpiexec -np 49 ./a.out 10 3360 && \
? time mpiexec -np 49 ./a.out 20 3360 && \
? time mpiexec -np 49 ./a.out 30 3360 && \
? time mpiexec -np 49 ./a.out 10 6720 && \
? time mpiexec -np 49 ./a.out 20 6720 && \
? time mpiexec -np 49 ./a.out 40 6720 && \
? time mpiexec -np 49 ./a.out 10 13440
gens 10 dim 840
0.152u 0.012s 0:02.40 6.6%      0+0k 0+0io 0pf+0w
gens 20 dim 840 timestamp;
0.152u 0.012s 0:02.56 6.2%      0+0k 0+0io 0pf+0w
gens 40 dim 840 lock();
0.144u 0.020s 0:02.54 6.2%      0+0k 0+0io 0pf+0w
gens 10 dim 1680 Elapsedtime(timestamp t){
0.144u 0.020s 0:02.44 6.5%      0+0k 0+0io 0pf+0w *
gens 20 dim 1680
0.140u 0.024s 0:02.53 6.3%      0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.148u 0.016s 0:02.94 5.1%      0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.148u 0.016s 0:03.11 4.8%      0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.132u 0.032s 0:03.67 4.3%      0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.132u 0.032s 0:04.31 3.7%      0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.136u 0.028s 0:05.12 2.9%      0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.124u 0.040s 0:06.80 2.3%      0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.140u 0.028s 0:11.05 1.4%      0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.148u 0.016s 0:12.48 1.2%      0+0k 0+0io 0pf+0w
```





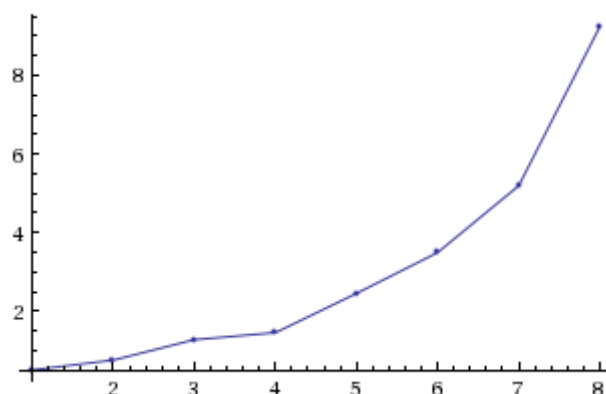
25 διεργασίες:

```
linux03:/home/users/std09080/Desktop/paral_koutou>time mpiexec -np 25 ./a.out 10 840 && \
? time mpiexec -np 25 ./a.out 20 840 && \
? time mpiexec -np 25 ./a.out 40 840 && \
? time mpiexec -np 25 ./a.out 10 1680 && \
? time mpiexec -np 25 ./a.out 20 1680 && \
? time mpiexec -np 25 ./a.out 30 1680 && \
? time mpiexec -np 25 ./a.out 10 3360 && \
? time mpiexec -np 25 ./a.out 20 3360 && \
? time mpiexec -np 25 ./a.out 30 3360 && \
? time mpiexec -np 25 ./a.out 10 6720 && \
? time mpiexec -np 25 ./a.out 20 6720 && \
? time mpiexec -np 25 ./a.out 40 6720 && \
? time mpiexec -np 25 ./a.out 10 13440
gens 10 dim 840
0.128u 0.032s 0:00.89 16.8%      0+0k 0+0io 0pf+0w
gens 20 dim 840 timestamp:
0.148u 0.012s 0:00.96 15.6%      0+0k 0+0io 0pf+0w
gens 40 dim 840 clock():
0.140u 0.020s 0:00.99 16.1%      0+0k 0+0io 0pf+0w
gens 10 dim 1680 Elapsedtime(timestamp t){
0.136u 0.024s 0:00.98 15.3%      0+0k 0+0io 0pf+0w * 1000.0f;
gens 20 dim 1680
0.156u 0.004s 0:01.18 12.7%      0+0k 0+0io 0pf+0w
gens 30 dim 1680
0.148u 0.012s 0:01.10 13.6%      0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.140u 0.020s 0:01.51 10.5%      0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.148u 0.012s 0:02.08 7.2%       0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.144u 0.012s 0:02.44 6.1%       0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.144u 0.016s 0:03.09 4.8%       0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.148u 0.012s 0:05.35 2.8%       0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.148u 0.012s 0:09.37 1.6%       0+0k 0+0io 0pf+0w
gens 10 dim 13440
0.152u 0.008s 0:10.69 1.4%       0+0k 0+0io 0pf+0w
```



1 διεργασίες:

```
linux03:/home/users/std09080/Desktop/paral_koutou>time mpiexec -np 1 ./a.out 10 840 && \
? time mpiexec -np 1 ./a.out 20 840 && \
? time mpiexec -np 1 ./a.out 40 840 && \
? time mpiexec -np 1 ./a.out 10 1680 && \
? time mpiexec -np 1 ./a.out 20 1680 && \
? time mpiexec -np 1 ./a.out 30 1680 && \
? time mpiexec -np 1 ./a.out 10 3360 && \
? time mpiexec -np 1 ./a.out 20 3360 && \
? time mpiexec -np 1 ./a.out 30 3360 && \
? time mpiexec -np 1 ./a.out 10 6720 && \
? time mpiexec -np 1 ./a.out 20 6720 && \
? time mpiexec -np 1 ./a.out 40 6720 && \
? time mpiexec -np 1 ./a.out 10 13440
gens 10 dim 840 of day(&tn, %d);
0.124u 0.032s 0:00.50 30.0% t.tv_0+0k 0+0io 0pf+0w (tn.tv_usec - t.tv_usec) / 1000000;
gens 20 dim 840
0.132u 0.020s 0:00.75 20.0% 0+0k 0+0io 0pf+0w
gens 40 dim 840
0.144u 0.012s 0:01.27 11.8% 0+0k 0+0io 0pf+0w
gens 10 dim 1680 getTimestamp(&tn){
0.140u 0.012s 0:01.45 10.3% 0+0k 0+0io 0pf+0w
gens 20 dim 1680
0.144u 0.012s 0:02.47 6.0% 0+0k 0+0io 0pf+0w
gens 30 dim 1680 (clock()-t) / 1000000;
0.132u 0.024s 0:03.49 4.2% 0+0k 0+0io 0pf+0w
gens 10 dim 3360
0.136u 0.020s 0:05.19 2.8% 0+0k 0+0io 0pf+0w
gens 20 dim 3360
0.144u 0.012s 0:09.24 1.6% 0+0k 0+0io 0pf+0w
gens 30 dim 3360
0.144u 0.012s 0:13.33 1.1% 0+0k 0+0io 0pf+0w
gens 10 dim 6720
0.144u 0.012s 0:20.13 0.7% 0+0k 0+0io 0pf+0w
gens 20 dim 6720
0.144u 0.012s 0:36.54 0.4% 0+0k 0+0io 0pf+0w
gens 40 dim 6720
0.156u 0.008s 1:09.12 0.2% 0+0k 0+0io 0pf+0w
gens 10 dim 13440
πάνω από 5 λεπτά...
```



## Cuda

Λίγα λόγια για το Cuda, υλοποιούμε τον δυσδιάστατο σαν μονοδιάστατο ,που για να γράψει σε κάθε γραμμή του, και στήλη του έχει κάποιο offset(συμπεριφέρεται δηλαδή ως προς τους κανόνες του game of life σαν δυσδιάστατος).

Αρχικά δημιουργούμε 3 πίνακες, 1 που κρατάει στην cpu το generation, και άλλα 2 στο gpu, το current\_gen, και το next\_gen.. Έχουμε 2 συναρτήσεις την GPU\_cycle, CPU\_cycle, στις οποίες, στην πρώτη(GPU\_cycle) τρέχουμε το πρόγραμμά μας, ανάλογα με το σπάσιμο που του κάνουμε κάθε φορά παράλληλα στην κάρτα γραφικών, και επιστρέφουμε τα αποτελέσματα που γράφονται στον πίνακά της cpu για το generation. Έπειτα καλούμε την συνάρτηση CPU\_cycle, και τρέχουμε αυτήν την φορά σειριακά την ίδια διαδικασία, μετρώντας τους χρόνους , και τυπώνοντας τους στην κονσόλα. Παρακάτω παραπέμπονται τα αποτελέσματα :

```
parallel: gens=5 dim=840    ms 111.713997
serial:  gens=5 dim=840    ms 119.049004

parallel: gens=5 dim=1680   ms 44.373001
serial:  gens=5 dim=1680   ms 499.472992

parallel: gens=20 dim=1680  ms 62.443970
serial:  gens=20 dim=1680  ms 1890.084961

parallel: gens=30 dim=1680  ms 73.942017
serial:  gens=30 dim=1680  ms 2842.375977

parallel: gens=50 dim=1680  ms 97.325989
serial:  gens=50 dim=1680  ms 4727.830078

parallel: gens=5 dim=3360   ms 59.532001
serial:  gens=5 dim=3360   ms 1897.754028
```

---



```
parallel: gens=20 dim=3360    ms 120.721008
serial:  gens=20 dim=3360    ms 7565.075195

parallel: gens=30 dim=3360    ms 159.962997
serial:  gens=30 dim=3360    ms 11382.353516

parallel: gens=50 dim=3360    ms 242.735992
serial:  gens=50 dim=3360    ms 18931.320312

parallel: gens=5 dim=6720     ms 117.629997
serial:  gens=5 dim=6720     ms 7591.837891

parallel: gens=20 dim=6720     ms 349.471008
serial:  gens=20 dim=6720     ms 30245.503906

parallel: gens=30 dim=6720     ms 512.299011
serial:  gens=30 dim=6720     ms 45358.492188

parallel: gens=50 dim=6720     ms 822.541992
serial:  gens=50 dim=6720     ms 75589.656250

parallel: gens=5 dim=13440     ms 352.394012
serial:  gens=5 dim=13440     ms 30243.117188

parallel: gens=20 dim=13440    ms 1278.213013
serial:  gens=20 dim=13440    ms 120929.804688

parallel: gens=30 dim=13440    ms 1895.962036
serial:  gens=30 dim=13440    ms 181389.953125

parallel: gens=50 dim=13440    ms 3143.235107
serial:  gens=50 dim=13440    ms 302294.937500

parallel: gens=5 dim=26880     ms 1314.226074
serial:  gens=5 dim=26880     ms 121152.546875

parallel: gens=20 dim=26880    ms 4962.763184
serial:  gens=20 dim=26880    ms 483771.031250

parallel: gens=30 dim=26880    ms 7501.439941
serial:  gens=30 dim=26880    ms 725618.000000
```