

# Задание 3

## Отчёт

### по реализации Игра «Жизнь»

Ши Хуэй shihuicollapsor@gmail.com

#### 1. Постановка задачи

Написать программу с использованием MPI, реализующую игру в жизнь.

1) Инициализировать начальное расположение живых клеток с помощью коллективных операций, метод инициализации произвольный. (например, можно сгенерировать начальные точки структур-глайдеров)

2) Провести K итераций игры

3) Начиная с K-ой итерации, если на каком-либо процессе число живых клеток на i-ой и i+1-ой итерациях игры совпадает, остановить игру на всей клеточной области, вывести общее число живых клеток в конце. Коммуникацию в данном случае также вести с помощью коллективных операций.

#### 2. Формат командной строки

```
mpic++ -std=c++11 -g -Wl,-rpath=/home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29/sem07  
mpiP-3.5 -L /home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29/sem07/mpiP-3.5 -lmpiP  
life_game_mpi.cpp -o work8
```

```
mpisubmit.pl -p 4 -w 00:05 ./work8
```

#### 3. Спецификация системы

- Operating system : Linux 6.8.0-45-generic
- Vendor string and code : GenuineIntel (1, 0x1)
- Model string and code : Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz (165, 0xa5)
- CPU revision : 2.0000000
- CPUID : Family/Model/Stepping 6/165/2, 0x06/0xa5/0x02
- CPU Max MHz : 5000
- CPU Min MHz : 800
- Total cores : 12
- SMT threads per core : 2
- Cores per socket : 6
- Sockets : 1
- Cores per NUMA region : 12
- NUMA regions : 1

- Running in a VM : no
- Number Hardware Counters : 10
- Max Multiplex Counters : 384
- Fast counter read (rdpmc): yes

## 5. Записи экспериментов и результаты

P=1

Total iterations times: 56

Total execution time: 0.0115825 seconds

P=2

Total iterations times: 56

Total execution time: 0.0086893 seconds

P=4

Total iterations times: 56

Total execution time: 0.00557945 seconds

P=8

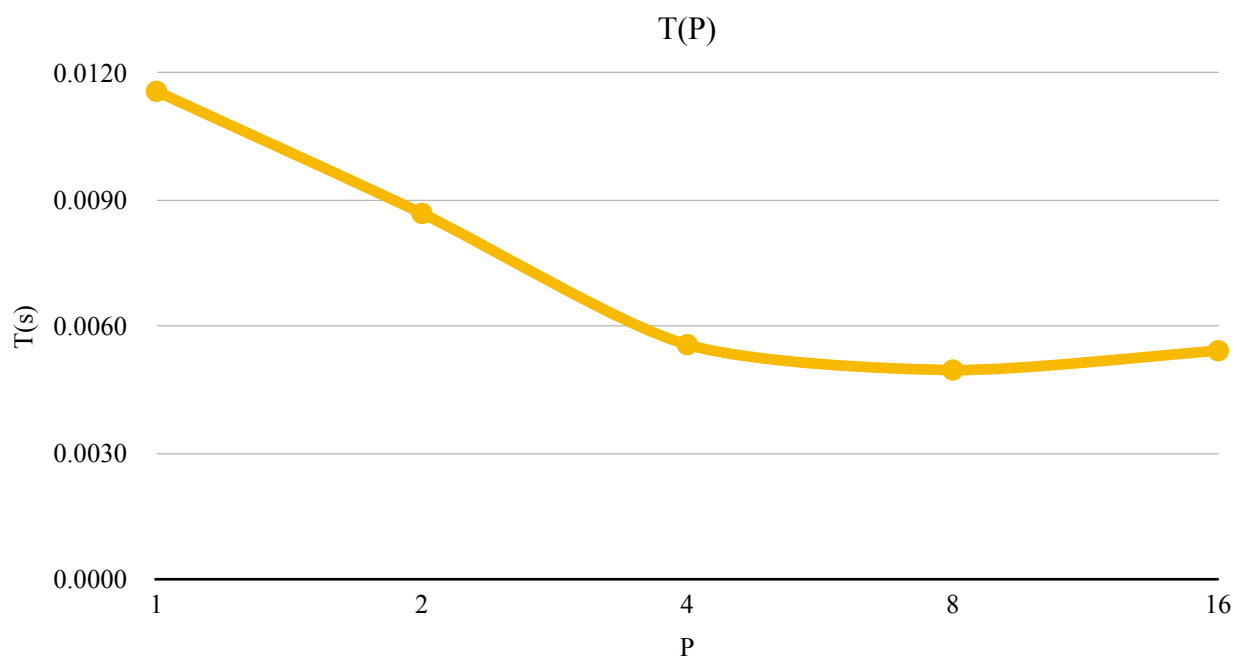
Total iterations times: 56

Total execution time: 0.00497826 seconds

P=16

Total iterations times: 56

Total execution time: 0.00543549 seconds



Поскольку слишком большой объем вывода во время выполнения кода приведет к увеличению накладных расходов, следующий дополнительный размер сетки равен  $32 * 32$ , а количество потоков равно 4. Выведите количество ячеек, сохраняющихся на каждой итерации.

*Current iterations: 1,Total alive cells: 5*  
*Current iterations: 2,Total alive cells: 5*  
*Current iterations: 3,Total alive cells: 5*  
*Current iterations: 4,Total alive cells: 5*  
*Current iterations: 5,Total alive cells: 5*  
*Current iterations: 6,Total alive cells: 5*  
*Current iterations: 7,Total alive cells: 5*  
*Current iterations: 8,Total alive cells: 5*  
*Current iterations: 9,Total alive cells: 5*  
*Current iterations: 10,Total alive cells: 5*  
*Current iterations: 11,Total alive cells: 5*  
*Current iterations: 12,Total alive cells: 5*  
*Current iterations: 13,Total alive cells: 5*  
*Current iterations: 14,Total alive cells: 5*  
*Current iterations: 15,Total alive cells: 5*  
*Current iterations: 16,Total alive cells: 5*  
*Current iterations: 17,Total alive cells: 5*  
*Current iterations: 18,Total alive cells: 5*  
*Current iterations: 19,Total alive cells: 5*  
*Current iterations: 20,Total alive cells: 5*  
*Current iterations: 21,Total alive cells: 5*  
*Current iterations: 22,Total alive cells: 5*  
*Current iterations: 23,Total alive cells: 5*  
*Current iterations: 24,Total alive cells: 5*  
*Current iterations: 25,Total alive cells: 5*  
*Current iterations: 26,Total alive cells: 5*  
*Current iterations: 27,Total alive cells: 5*  
*Current iterations: 28,Total alive cells: 5*  
*Current iterations: 29,Total alive cells: 5*  
*Current iterations: 30,Total alive cells: 5*  
*Current iterations: 31,Total alive cells: 5*  
*Current iterations: 32,Total alive cells: 5*  
*Current iterations: 33,Total alive cells: 5*

Current iterations: 34,Total alive cells: 5  
Current iterations: 35,Total alive cells: 5  
Current iterations: 36,Total alive cells: 5  
Current iterations: 37,Total alive cells: 5  
Current iterations: 38,Total alive cells: 5  
Current iterations: 39,Total alive cells: 5  
Current iterations: 40,Total alive cells: 5  
Current iterations: 41,Total alive cells: 5  
Current iterations: 42,Total alive cells: 5  
Current iterations: 43,Total alive cells: 5  
Current iterations: 44,Total alive cells: 5  
Current iterations: 45,Total alive cells: 5  
Current iterations: 46,Total alive cells: 5  
Current iterations: 47,Total alive cells: 5  
Current iterations: 48,Total alive cells: 5  
Current iterations: 49,Total alive cells: 5  
Current iterations: 50,Total alive cells: 5  
Current iterations: 51,Total alive cells: 5  
Current iterations: 52,Total alive cells: 5  
Current iterations: 53,Total alive cells: 4  
Current iterations: 54,Total alive cells: 3  
Current iterations: 55,Total alive cells: 4

*[edu-cmc-sqi22-29@polus-ib sem08]\$ cat ./work8.4.80843.1.mpiP*

*@ mpiP*

*@ Command : ./work8*

*@ Version : 3.5.0*

*@ MPIP Build date : Nov 6 2024, 21:55:54*

*@ Start time : 2024 11 21 15:53:55*

*@ Stop time : 2024 11 21 15:53:55*

*@ Timer Used : gettimeofday*

*@ MPIP env var : [null]*

*@ Collector Rank : 0*

*@ Collector PID : 80843*

*@ Final Output Dir : .*

*@ Report generation : Single collector task*

@ MPI Task Assignment : 0 polus-c2-ib.bmc.hpc.cs.msu.ru  
 @ MPI Task Assignment : 1 polus-c2-ib.bmc.hpc.cs.msu.ru  
 @ MPI Task Assignment : 2 polus-c2-ib.bmc.hpc.cs.msu.ru  
 @ MPI Task Assignment : 3 polus-c2-ib.bmc.hpc.cs.msu.ru

-----  
 @--- MPI Time (seconds) -----  
 -----

Task	AppTime	MPITime	MPI%
0	0.00563	0.00221	39.21
1	0.00525	0.00121	23.00
2	0.00561	0.0026	46.29
3	0.00525	0.00132	25.15
*	0.0217	0.00733	33.72

-----  
 @--- Callsites: 9 -----  
 -----

ID	Lev	File/Address	Line	Parent_Funct	MPI_Call
1	0	0x1000000b297c		[unknown]	Gatherv
2	0	0x1000000bb06c		[unknown]	Waitall
3	0	0x1000000b7f6c		[unknown]	Scatterv
4	0	0x1000000ae44c		[unknown]	Bcast
5	0	0x1000000b6c2c		[unknown]	Reduce
6	0	0x1000000b5c2c		[unknown]	Isend
7	0	0x1000000b566c		[unknown]	Irecv
8	0	0x1000000b5c2c		[unknown]	Isend
9	0	0x1000000b566c		[unknown]	Irecv

-----  
 @--- Aggregate Time (top twenty, descending, milliseconds) -----  
 -----

Call	Site	Time	App%	MPI%	Count	COV
Reduce	5	3	13.80	40.92	224	0.80
Gatherv	1	1.7	7.81	23.16	224	0.42
Waitall	2	1.13	5.21	15.44	224	0.76
Bcast	4	0.849	3.90	11.58	224	0.33
Isend	8	0.236	1.09	3.22	168	0.15

<i>Isend</i>	6	0.181	0.83	2.47	168	0.35
<i>Scatterv</i>	3	0.135	0.62	1.84	4	0.23
<i>Irecv</i>	9	0.068	0.31	0.93	168	0.07
<i>Irecv</i>	7	0.033	0.15	0.45	168	0.42

-----  
@--- Aggregate Sent Message Size (top twenty, descending, bytes) -----  
-----

<i>Call</i>	<i>Site</i>	<i>Count</i>	<i>Total</i>	<i>Avrg</i>	<i>Sent%</i>
<i>Gatherv</i>	1	224	2.29e+05	1.02e+03	83.66
<i>Isend</i>	6	168	2.15e+04	128	7.84
<i>Isend</i>	8	168	2.15e+04	128	7.84
<i>Reduce</i>	5	224	896	4	0.33
<i>Bcast</i>	4	224	896	4	0.33

-----  
@--- Callsite Time statistics (all, milliseconds): 32 -----  
-----

<i>Name</i>	<i>Site</i>	<i>Rank</i>	<i>Count</i>	<i>Max</i>	<i>Mean</i>	<i>Min</i>	<i>App%</i>	<i>MPI%</i>
<i>Bcast</i>	4	0	56	0.01	0.002	0.001	1.99	5.07
<i>Bcast</i>	4	1	56	0.02	0.00443	0.002	4.72	20.53
<i>Bcast</i>	4	2	56	0.017	0.00391	0.002	3.90	8.43
<i>Bcast</i>	4	3	56	0.02	0.00482	0.002	5.14	20.44
<i>Bcast</i>	4	*	224	0.02	0.00379	0.001	3.90	11.58

<i>Gatherv</i>	1	0	56	0.021	0.00293	0.002	2.91	7.43
<i>Gatherv</i>	1	1	56	0.019	0.00882	0.008	9.40	40.89
<i>Gatherv</i>	1	2	56	0.026	0.00998	0.009	9.96	21.52
<i>Gatherv</i>	1	3	56	0.019	0.00859	0.008	9.16	36.41
<i>Gatherv</i>	1	*	224	0.026	0.00758	0.002	7.81	23.16

<i>Irecv</i>	7	0	56	0.004	0.000286	0	0.28	0.72
<i>Irecv</i>	7	1	56	0.001	0.000125	0	0.13	0.58
<i>Irecv</i>	7	2	56	0.001	0.000179	0	0.18	0.39
<i>Irecv</i>	7	*	168	0.004	0.000196	0	0.15	0.45

<i>Irecv</i>	9	1	56	0.009	0.000429	0	0.46	1.99
<i>Irecv</i>	9	2	56	0.007	0.000411	0	0.41	0.89

<i>Irecv</i>	9	3	56	0.009	0.000375	0	0.40	1.59
<i>Irecv</i>	9	*	168	0.009	0.000405	0	0.31	0.93
<i>Isend</i>	6	0	56	0.009	0.000643	0	0.64	1.63
<i>Isend</i>	6	1	56	0.011	0.00129	0	1.37	5.96
<i>Isend</i>	6	2	56	0.013	0.0013	0	1.30	2.81
<i>Isend</i>	6	*	168	0.013	0.00108	0	0.83	2.47
<i>Isend</i>	8	1	56	0.014	0.00118	0	1.26	5.46
<i>Isend</i>	8	2	56	0.014	0.00145	0	1.44	3.12
<i>Isend</i>	8	3	56	0.015	0.00159	0	1.69	6.74
<i>Isend</i>	8	*	168	0.015	0.0014	0	1.09	3.22
<i>Reduce</i>	5	0	56	0.075	0.0246	0.019	24.43	62.30
<i>Reduce</i>	5	1	56	0.04	0.00252	0.001	2.68	11.67
<i>Reduce</i>	5	2	56	0.064	0.0204	0.015	20.36	43.97
<i>Reduce</i>	5	3	56	0.052	0.00613	0.002	6.53	25.97
<i>Reduce</i>	5	*	224	0.075	0.0134	0.001	13.80	40.92
<i>Scatterv</i>	3	0	1	0.025	0.025	0.025	0.44	1.13
<i>Scatterv</i>	3	1	1	0.042	0.042	0.042	0.80	3.48
<i>Scatterv</i>	3	2	1	0.03	0.03	0.03	0.53	1.16
<i>Scatterv</i>	3	3	1	0.038	0.038	0.038	0.72	2.88
<i>Scatterv</i>	3	*	4	0.042	0.0338	0.025	0.62	1.84
<i>Waitall</i>	2	0	56	0.342	0.00855	0	8.51	21.70
<i>Waitall</i>	2	1	56	0.047	0.00204	0	2.17	9.44
<i>Waitall</i>	2	2	56	0.351	0.00821	0	8.20	17.71
<i>Waitall</i>	2	3	56	0.009	0.00141	0	1.50	5.98
<i>Waitall</i>	2	*	224	0.351	0.00505	0	5.21	15.44

-----  
 @--- Callsite Message Sent statistics (all, sent bytes) -----  
 -----

<i>Name</i>	<i>Site</i>	<i>Rank</i>	<i>Count</i>	<i>Max</i>	<i>Mean</i>	<i>Min</i>	<i>Sum</i>
<i>Bcast</i>	4	0	56	4	4	4	224
<i>Bcast</i>	4	1	56	4	4	4	224

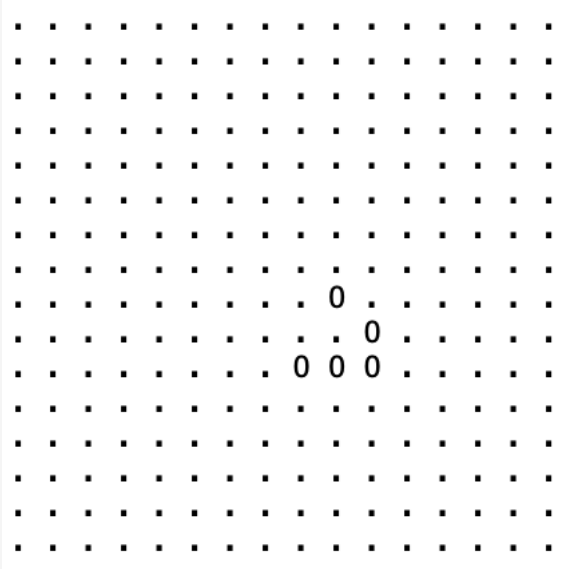
<i>Bcast</i>	4	2	56	4	4	4	224
<i>Bcast</i>	4	3	56	4	4	4	224
<i>Bcast</i>	4	*	224	4	4	4	896
<i>Gatherv</i>	1	0	56	1024	1024	1024	5.734e+04
<i>Gatherv</i>	1	1	56	1024	1024	1024	5.734e+04
<i>Gatherv</i>	1	2	56	1024	1024	1024	5.734e+04
<i>Gatherv</i>	1	3	56	1024	1024	1024	5.734e+04
<i>Gatherv</i>	1	*	224	1024	1024	1024	2.294e+05
<i>Isend</i>	6	0	56	128	128	128	7168
<i>Isend</i>	6	1	56	128	128	128	7168
<i>Isend</i>	6	2	56	128	128	128	7168
<i>Isend</i>	6	*	168	128	128	128	2.15e+04
<i>Isend</i>	8	1	56	128	128	128	7168
<i>Isend</i>	8	2	56	128	128	128	7168
<i>Isend</i>	8	3	56	128	128	128	7168
<i>Isend</i>	8	*	168	128	128	128	2.15e+04
<i>Reduce</i>	5	0	56	4	4	4	224
<i>Reduce</i>	5	1	56	4	4	4	224
<i>Reduce</i>	5	2	56	4	4	4	224
<i>Reduce</i>	5	3	56	4	4	4	224
<i>Reduce</i>	5	*	224	4	4	4	896

-----  
 @--- End of Report -----

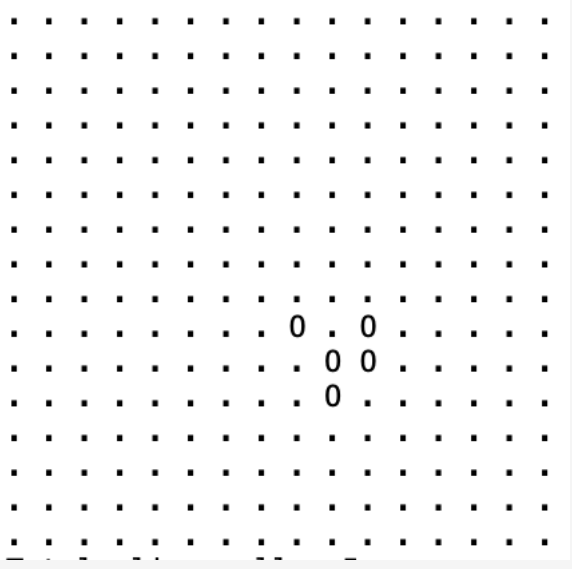
Ниже приведена еще пара примеров, когда размер сетки равен 16 \* 16, а количество потоков равно 4. Ячейки в сетке визуальнo отображаются на каждой итерации.

Initial Grid State:	Grid State at Iteration 1:
---------------------	----------------------------

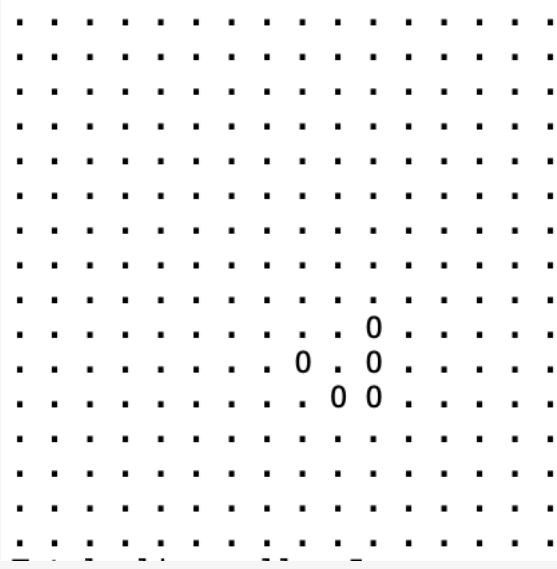




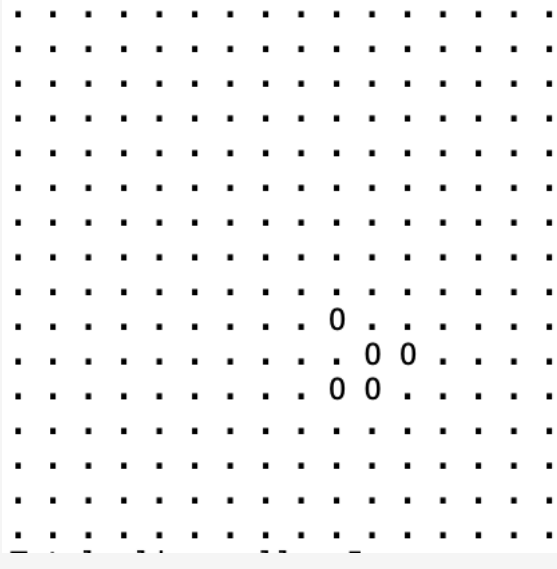
Grid State at Iteration 2:



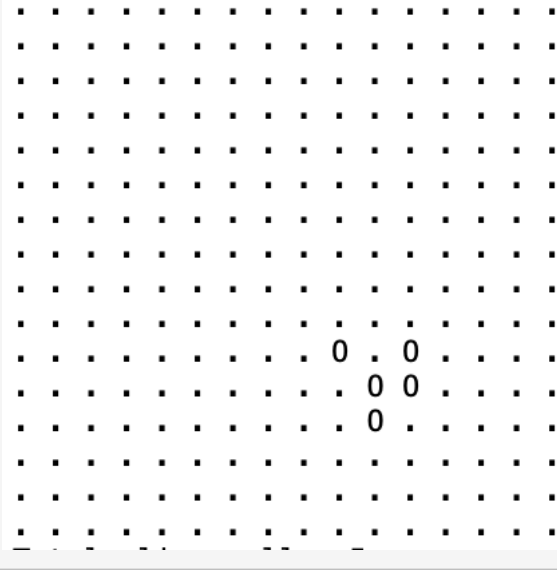
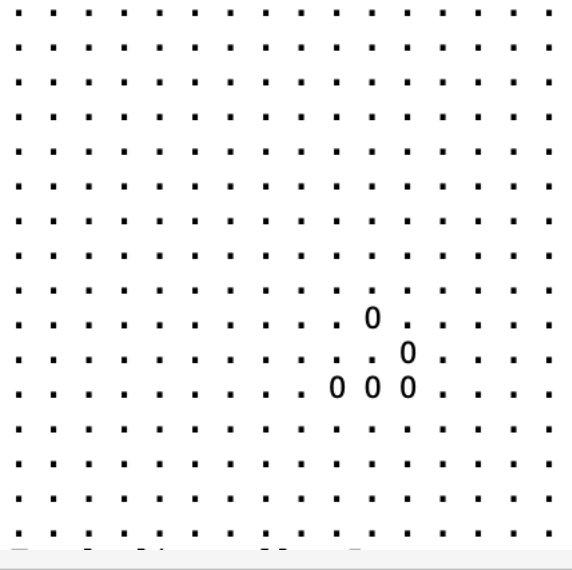
Grid State at Iteration 3:

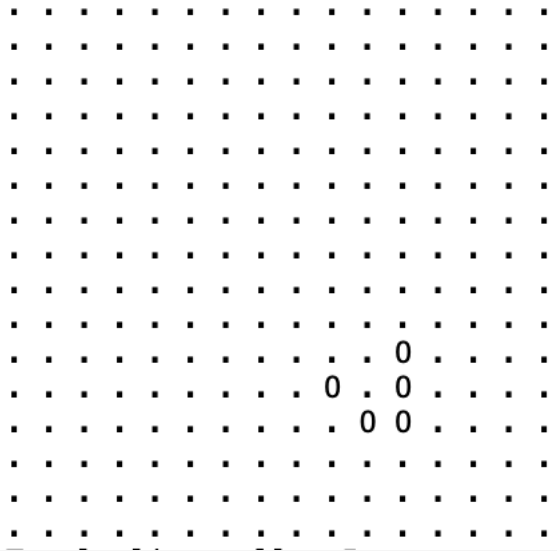
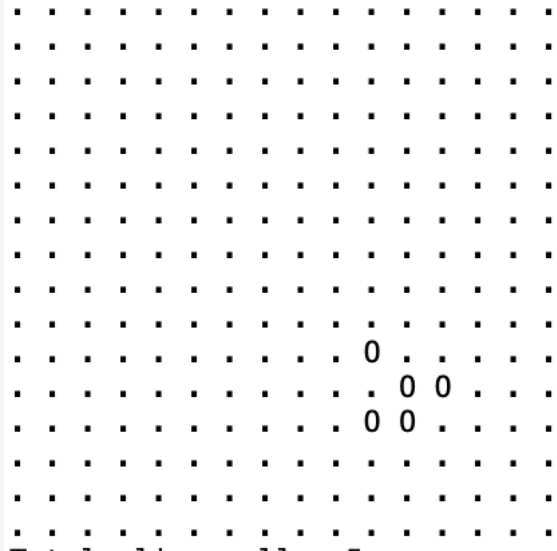
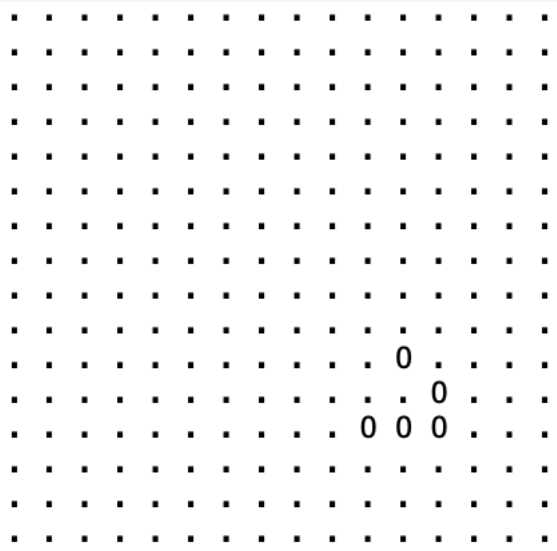
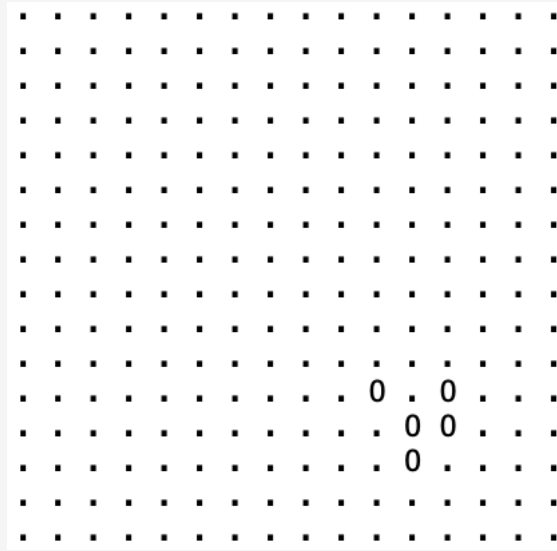
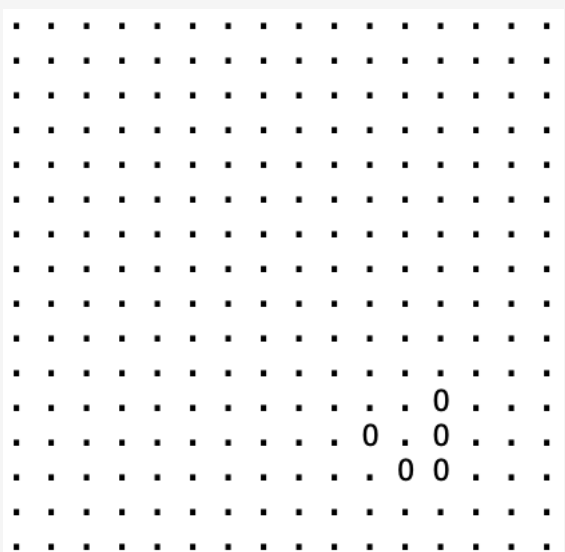
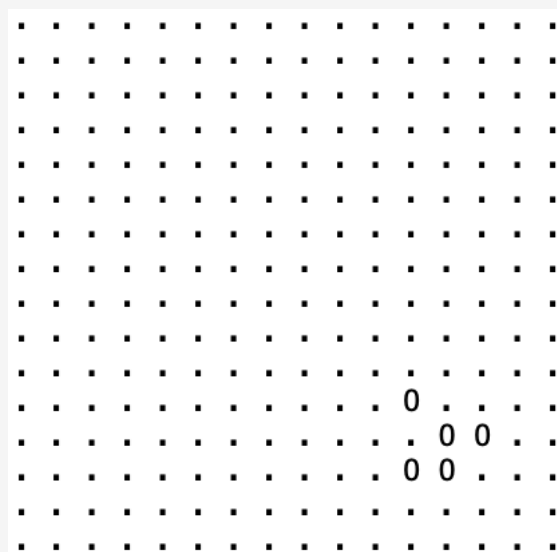


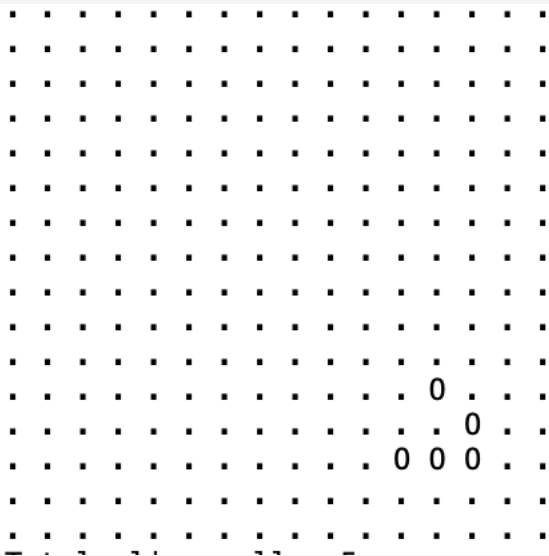
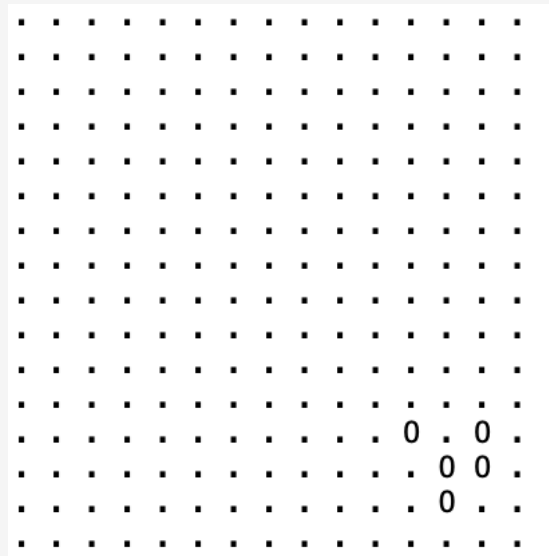
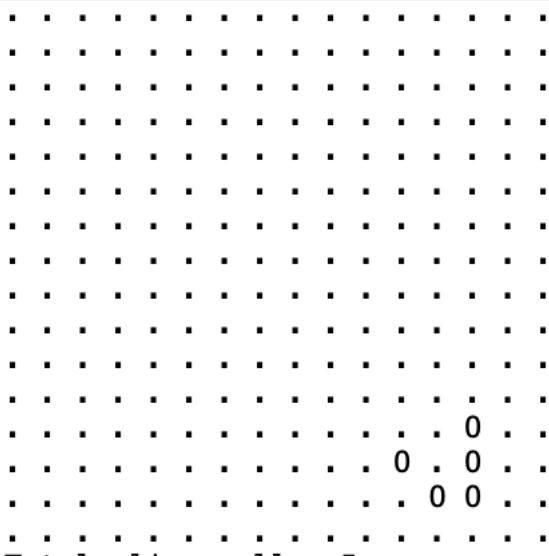
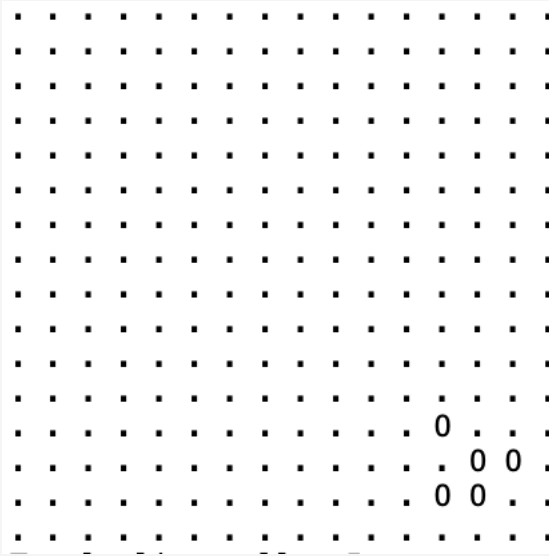
Grid State at Iteration 4:

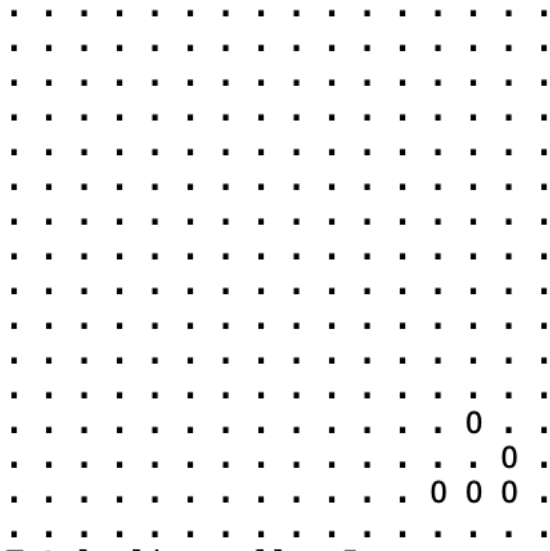


Grid State at Iteration 5:

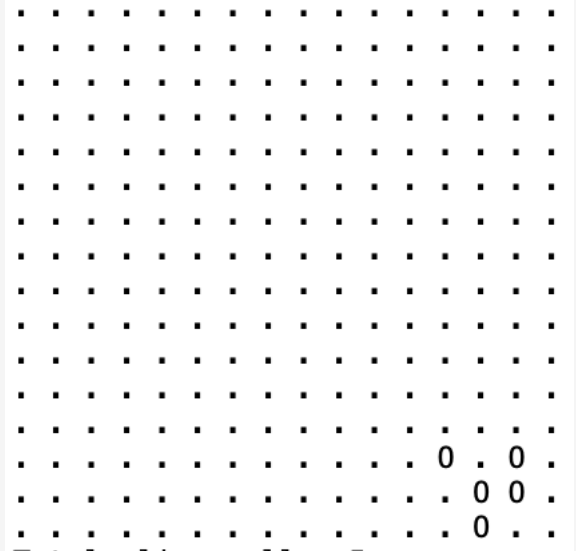


Grid State at Iteration 6:	Grid State at Iteration 7:
	
Grid State at Iteration 8:	Grid State at Iteration 9:
	
Grid State at Iteration 10:	Grid State at Iteration 11:
	

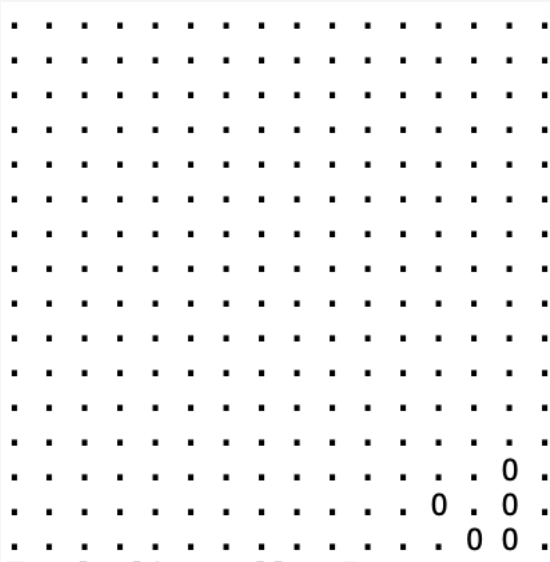
Grid State at Iteration 12:	Grid State at Iteration 13:
	
Grid State at Iteration 14:	Grid State at Iteration 15:
	
Grid State at Iteration 16:	Grid State at Iteration 17:



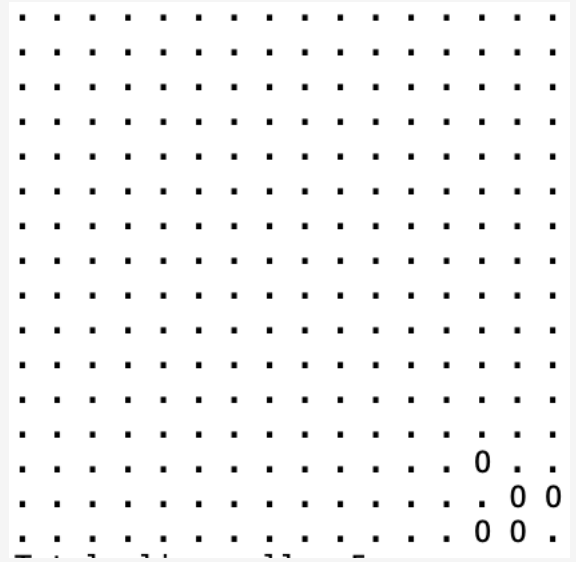
Grid State at Iteration 18:



Grid State at Iteration 19:



Grid State at Iteration 20:



Grid State at Iteration 21:

