

Измерение производительности (кол-во операций в миллисекунду) различных потокобезопасных счетчиков.

Опыт1: кол-во потоков на чтение равно кол-ву потоков на запись.

Action (ops/ms)		1+1	2+2	4+4	8+8	16+16
MutexCounter (synchronized)	read	9533	8230	6341	6289	5649
	write	9498	10936	12581	12458	13269
LockCounter	read	8500	8816	9283	10948	10484
	write	8354	16641	15463	14166	12910
ConcurrentCounter (atomic)	read	921306	1791366	18038861	1709810	1697363
	write	110592	47551	60775	55150	56695

Вывод:

- 1) **MutexCounter** - с увеличением кол-ва потоков число операций чтения уменьшается, а записи растут
- 2) **LockCounter** - с увеличением количества потоков число операций чтения растет, записи увеличивается в 2 раза после перехода на 2 потока, а затем медленно уменьшается
- 3) **ConcurrentCounter** - самый производительный счетчик, количество операций чтения больше чем операций записи.

Опыт2: чтение и запись тестируются отдельно

Action (ops/ms)		1	2	4	8	16
MutexCounter (synchronized)	read	46813	22706	21320	22671	21975
	write	47759	20203	22647	22596	19679
LockCounter	read	56142	7796	33244	35321	35248
	write	46862	7211	28889	28364	28818
ConcurrentCounter (atomic)	read	1305100	2580154	4866908	4559940	4827313
	write	187774	47526	39847	35982	37259

Вывод:

- 1) **MutexCounter** - при распараллеливании уменьшается производительность в 2 раза, параллельная производительность 22тыс операций и не зависит от кол-ва потоков
- 2) **LockCounter** - при распараллеливании уменьшается производительность, при двух потоках самая низкая производительность, при остальных - постоянная.
- 3) **ConcurrentCounter** - самый производительный счетчик, производительность записи больше чтения, производительность чтения падает при увеличении кол-ва потоков, запись - растет.

Опыт3: один поток на чтение (READ_THREAD_COUNT=1), на запись - 1,2,3,4

Action (ops/ms)		1	2	3	4
MutexCounter (synchronized)	read	10123	6880	5701	4569
	write	10090	14266	15091	17577
LockCounter	read	4522	10369	5037	1440
	write	4001	20446	24810	28679
ConcurrentCounter (atomic)	read	1283975	1161911	1186352	1067533
	write	186057	47424	48179	48052

Вывод:

- 1) MutexCounter - чем больше потоков, тем больше производительность записи, но чтение падает, общая производительность 20-22тыс операций и не зависит от кол-ва потоков.
- 2) LockCounter - при увеличении потоков производительность записи увеличивается.
- 3) ConcurrentCounter - самый производительный счетчик, производительность записи падает до 48тыс операций и остается постоянной для кол-ва потоков = 2,3,4

Опыт4: один поток на запись (WRITE_THREAD_COUNT=1), на чтение - 1,2,3,4
write=1

Action (ops/ms)		1	2	3	4
MutexCounter (synchronized)	read	10763	14947	17017	15947
	write	11419	7987	5909	3957
LockCounter	read	4194	19560	21236	28586
	write	3509	13286	12007	7370
ConcurrentCounter (atomic)	read	1284999	2489973	3661125	3915124
	write	186505	181035	179607	157956

Вывод:

- 1) MutexCounter - чем больше потоков, тем больше производительность чтения, но запись падает, общая производительность 20-22тыс операций и не зависит от кол-ва потоков.
- 2) LockCounter - при увеличении потоков производительность чтения увеличивается.
- 3) ConcurrentCounter - самый производительный счетчик, производительность чтения растет, записи - падает.