1000 x 1000 board 10 000 rectangles

1 thread:
- 51605
- 51321
- 50510
- 53428
- 50576

average: 51488

2 threads:
- 46565
- 48119
- 47998
- 48869
- 49317

average: 48173

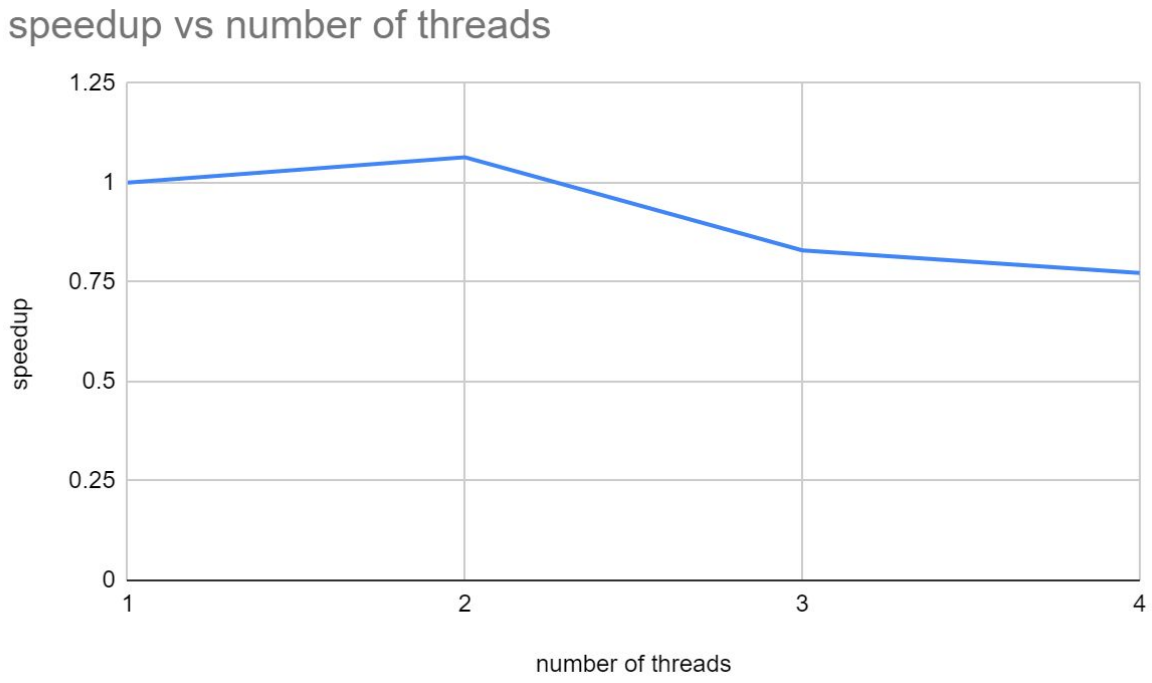
3 threads:
- 62867
- 62749
- 55867
- 63278
- 56331

average: 60218

4 threads:
- 62535
- 61419
- 62052
- 66497
- 63455

average: 63191

speedup vs number of threads:
1 thread: 1
2 threads: 1.064
3 threads: 0.83
4 threads: 0.773



Explanation: I think the reason why the speedup drops at 3 and 4 threads is because the threads are getting the lock and then check if the needed pixels are free in the image. This could cause some threads to keep getting the lock while the image is not free for them. Because of this, other threads that could actually do work are blocked since they may not get the lock. This is especially apparent when the number of threads are high. To solve this problem, I could try to check if the image is free before getting the lock, or implement a queue to make sure every thread has access to the lock fairly.