Homework 1 Milestone 2 Report

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

Student:

Kushagra Goel

Penn ID

Kgoel96

Course:

CIS 455/555 Internet and Web Systems

Testing Scenario

A route was created to calculate number of prime numbers upto a given number which can be provided as the part of the URI (for eg localhost:8080/benchmark/10000).

Two different levels of testing were done

- Moderate load
  - Prime Numbers upto 100 with 1000 simultaneous requests
- Large load
  - Prime Numbers upto 1000 with 100 simultaneous requests
- Thread Sleep Test
  - Each thread sleeps for 10 ms, 1000 simultaneous requests

Then using apache bench 100 requests 100 at a time, were made and statistics are recorded for analysis. The threadPool size was varied in order to benchmark.

Here is the registered route:

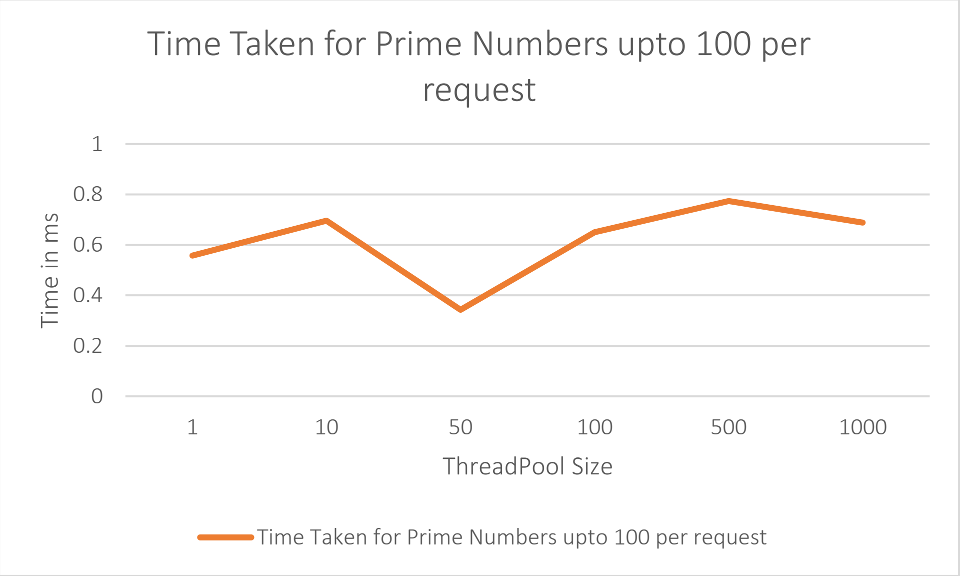
```
  get("/benchmark/:number", (request,response) -> {
        long value = Long.valueOf(request.params("number"));
        long count = 0;
        for(long i = 2 ; i <= value; i++) {
                boolean isPrime = true;
                for(long j = 2; j <= value; j++) {
                        if( i == j)continue;
                        if(i%j == 0) {
                                isPrime = false;
                        }
                }
                if(isPrime)count++;
        }
        return "<!DOCTYPE html><html><body><p>There are " + String.valueOf(count) + "
prime numbers upto " + String.valueOf(value) + "</body></html>";
  });
```

Route Number of Prime Numbers

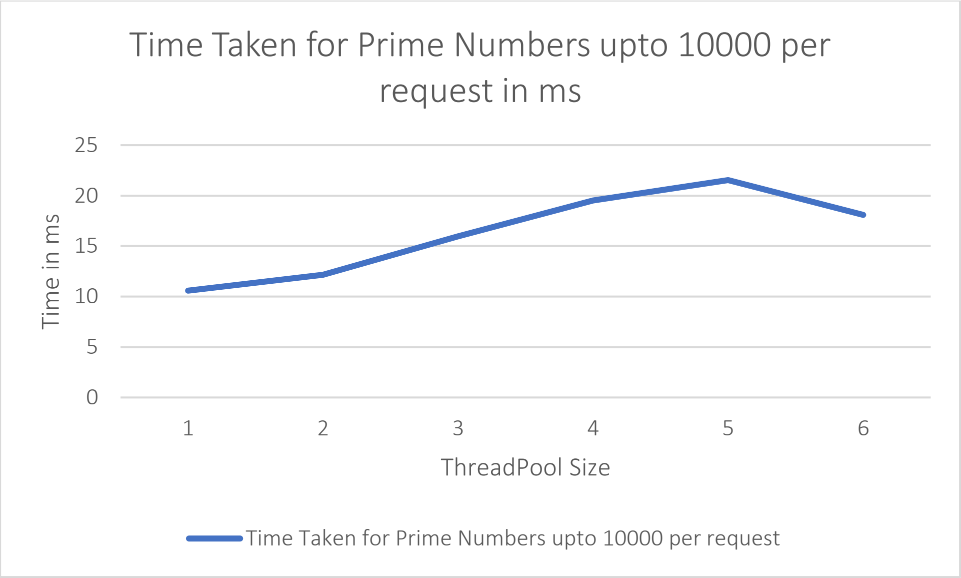
Results

Thread Pool Size	Time Taken for Prime Numbers upto 100 per request	Time Taken for Prime Numbers upto 10000 per request	Thread Sleep Test total time
1	0.558	10.587	11073.544
10	0.696	12.148	1293.992
50	0.343	15.982	1204.436
100	0.651	19.512	1167.599
500	0.774	21.539	1050.238
1000	0.689	18.086	1073.94

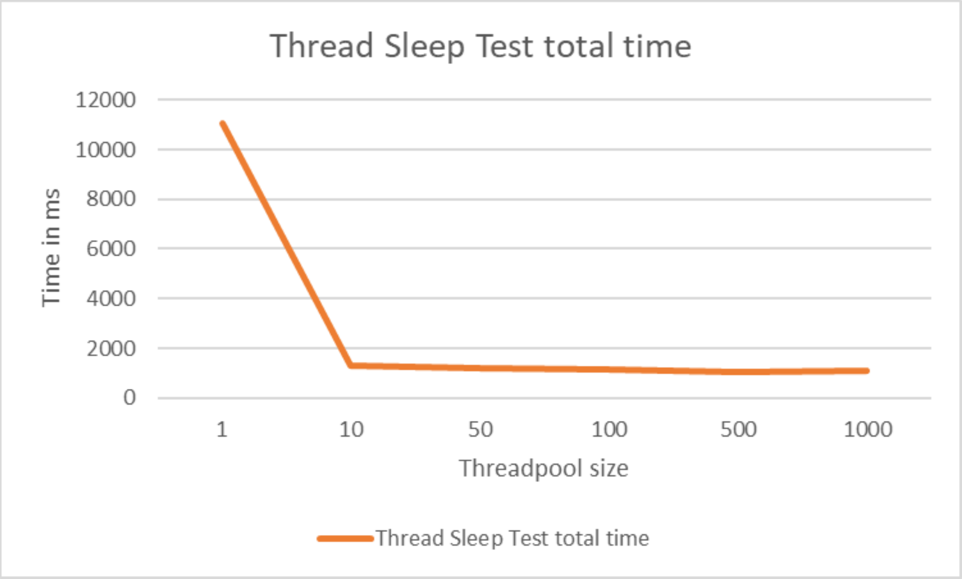
- Moderate Load (mean time per request)



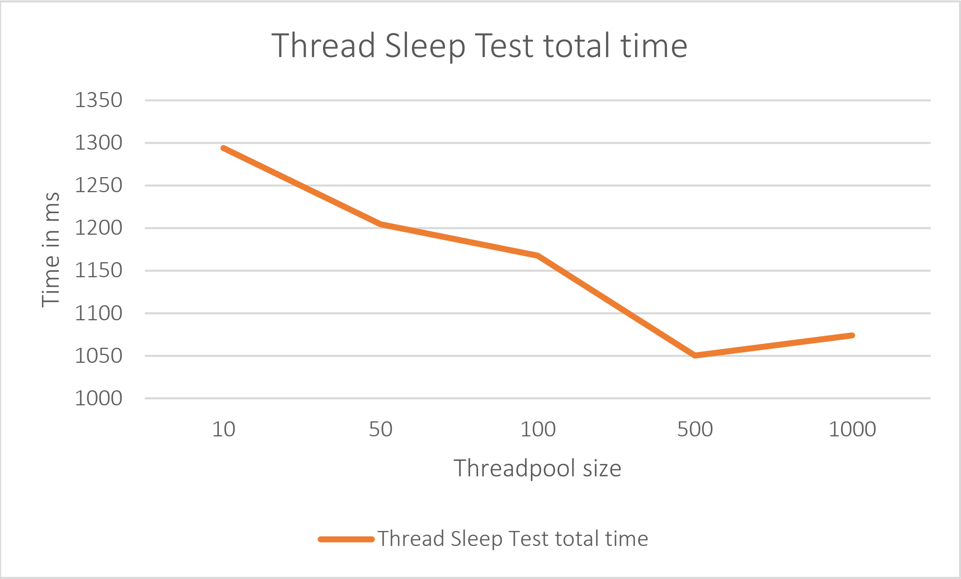
- Large Load (mean time per request)



- Thread Sleep Test (total time)
  - Including Single thread



- Excluding Single thread



Conclusion

We observe that for the prime number test, more threads increased total time taken per request. This might be due to the fact that the prime test was very computationally dense and therefore the overhead of context switching affected it negatively.

Whereas for the thread sleep test, which was designed to simulate blocking calls, we see that more threads reduce the total time taken. In fact there is a huge drop as the server goes from single thread to multiple thread.

Finally, as to how many threads are best, it depends on the type of workload the server is going to handle. Since in our case, mostly it is file I/O (MS1), 100 threads is a fair tradeoff between the two different test scenarios presented.