Assignment 3

Aryaman Srivastava 2210110206

Validating Sudoku Solution

By following all the given instructions, we get the following results:

Experiment 1(Keeping size changing, threads constant**):**

Single threading:

4*4 Input -

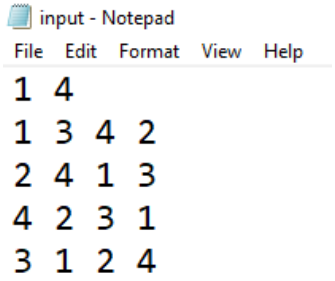
1 4

1 3 4 2

2 4 1 3

4 2 3 1

3 1 2 4



4*4 Output –

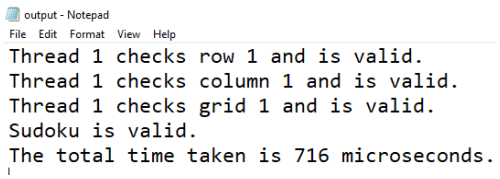
Thread 1 checks row 1 and is valid.

Thread 1 checks column 1 and is valid.

Thread 1 checks grid 1 and is valid.

Sudoku is valid.

The total time taken is 716 microseconds.



9*9 Input -

1 9

5 3 4 6 7 8 9 1 2

6 7 2 1 9 5 3 4 8

1 9 8 3 4 2 5 6 7

8 5 9 7 6 1 4 2 3

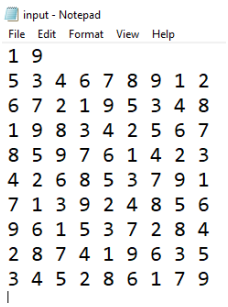
4 2 6 8 5 3 7 9 1

7 1 3 9 2 4 8 5 6

9 6 1 5 3 7 2 8 4

2 8 7 4 1 9 6 3 5

3 4 5 2 8 6 1 7 9



9*9 Output –

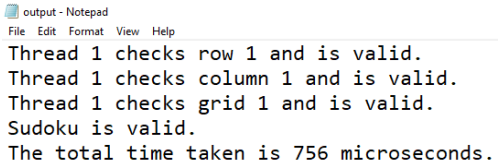
Thread 1 checks row 1 and is valid.

Thread 1 checks column 1 and is valid.

Thread 1 checks grid 1 and is valid.

Sudoku is valid.

The total time taken is 756 microseconds.



16*16 Input -

1 16

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

11 12 13 14 15 16 1 2 3 4 5 6 7 8 9 10

5 6 7 8 9 10 11 12 13 14 15 16 1 2 3 4

9 10 11 12 13 14 15 16 1 2 3 4 5 6 7 8

3 4 5 6 7 8 9 10 11 12 13 14 15 16 1 2

7 8 9 10 11 12 13 14 15 16 1 2 3 4 5 6

12 13 14 15 16 1 2 3 4 5 6 7 8 9 10 11

16 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1

8 9 10 11 12 13 14 15 16 1 2 3 4 5 6 7

4 5 6 7 8 9 10 11 12 13 14 15 16 1 2 3

13 14 15 16 1 2 3 4 5 6 7 8 9 10 11 12

6 7 8 9 10 11 12 13 14 15 16 1 2 3 4 5

10 11 12 13 14 15 16 1 2 3 4 5 6 7 8 9

15 16 1 2 3 4 5 6 7 8 9 10 11 12 13 14

9 10 11 12 13 14 15 16 1 2 3 4 5 6 7 8

16*16 Output –

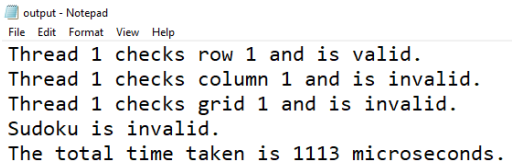
Thread 1 checks row 1 and is valid.

Thread 1 checks column 1 and is invalid.

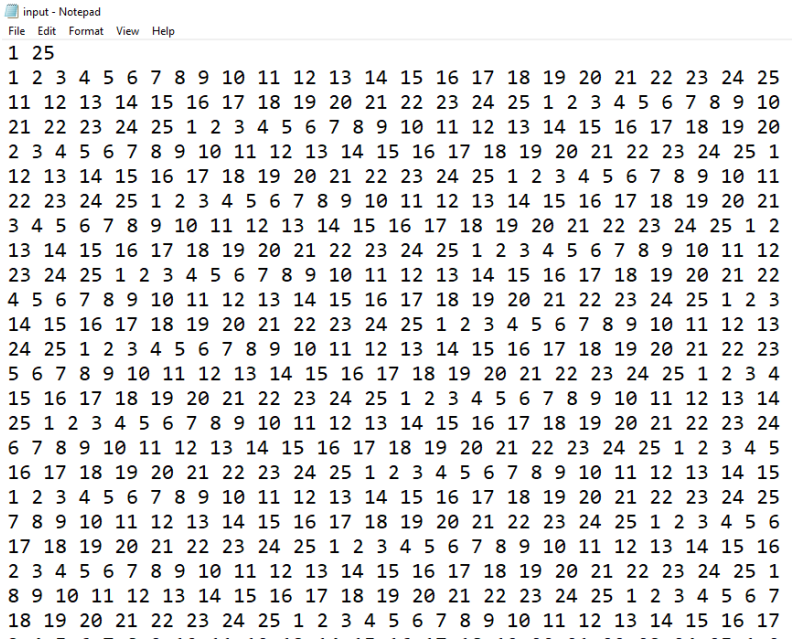
Thread 1 checks grid 1 and is invalid.

Sudoku is invalid.

The total time taken is 1113 microseconds.



25*25 Input -



25*25 Output –

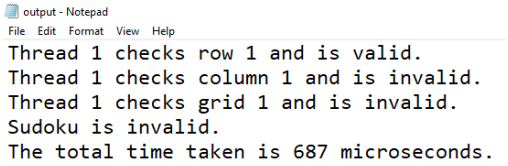
Thread 1 checks row 1 and is valid.

Thread 1 checks column 1 and is invalid.

Thread 1 checks grid 1 and is invalid.

Sudoku is invalid.

The total time taken is 687 microseconds.



Pthread:

4*4 Input –

16 4

1 3 4 2

2 4 1 3

4 2 3 1

3 1 2 4

input - Notepad

File Edit Format View Help

```
16 4
1 3 4 2
2 4 1 3
4 2 3 1
3 1 2 4
```

4*4 Output –

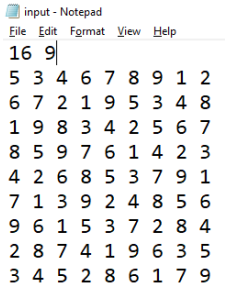
output - Notepad

File Edit Format View Help

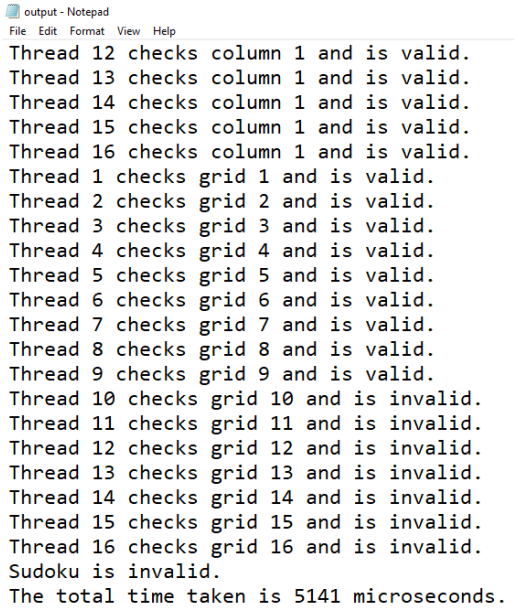
```
Thread 12 checks column 1 and is valid.
Thread 13 checks column 1 and is valid.
Thread 14 checks column 1 and is valid.
Thread 15 checks column 1 and is valid.
Thread 16 checks column 1 and is valid.
Thread 1 checks grid 1 and is valid.
Thread 2 checks grid 2 and is valid.
Thread 3 checks grid 3 and is valid.
Thread 4 checks grid 4 and is valid.
Thread 5 checks grid 5 and is invalid.
Thread 6 checks grid 6 and is invalid.
Thread 7 checks grid 7 and is invalid.
Thread 8 checks grid 8 and is invalid.
Thread 9 checks grid 9 and is valid.
Thread 10 checks grid 10 and is valid.
Thread 11 checks grid 11 and is valid.
Thread 12 checks grid 12 and is valid.
Thread 13 checks grid 13 and is invalid.
Thread 14 checks grid 14 and is invalid.
Thread 15 checks grid 15 and is invalid.
Thread 16 checks grid 16 and is invalid.
Sudoku is invalid.
The total time taken is 11876 microseconds.
```

9*9 Input –

```
16 9

5 3 4 6 7 8 9 1 2

6 7 2 1 9 5 3 4 8

1 9 8 3 4 2 5 6 7

8 5 9 7 6 1 4 2 3

4 2 6 8 5 3 7 9 1

7 1 3 9 2 4 8 5 6

9 6 1 5 3 7 2 8 4

2 8 7 4 1 9 6 3 5

3 4 5 2 8 6 1 7 9
```



9*9 Input –



16*16 Input –

16 16

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

11 12 13 14 15 16 1 2 3 4 5 6 7 8 9 10

5 6 7 8 9 10 11 12 13 14 15 16 1 2 3 4

9 10 11 12 13 14 15 16 1 2 3 4 5 6 7 8

3 4 5 6 7 8 9 10 11 12 13 14 15 16 1 2

7 8 9 10 11 12 13 14 15 16 1 2 3 4 5 6

12 13 14 15 16 1 2 3 4 5 6 7 8 9 10 11

16 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1

8 9 10 11 12 13 14 15 16 1 2 3 4 5 6 7

4 5 6 7 8 9 10 11 12 13 14 15 16 1 2 3

13 14 15 16 1 2 3 4 5 6 7 8 9 10 11 12

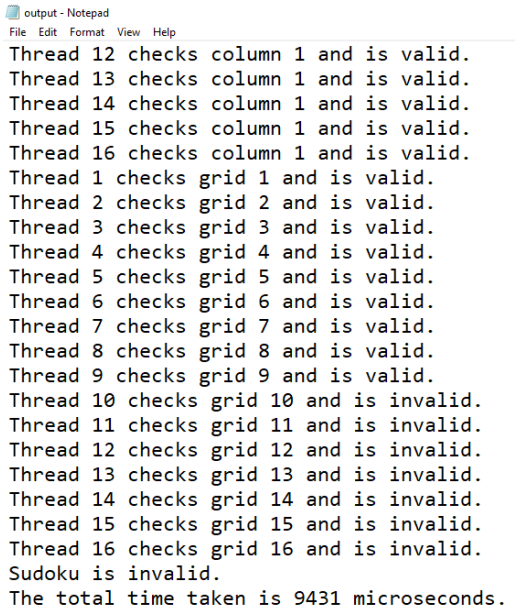
6 7 8 9 10 11 12 13 14 15 16 1 2 3 4 5

10 11 12 13 14 15 16 1 2 3 4 5 6 7 8 9

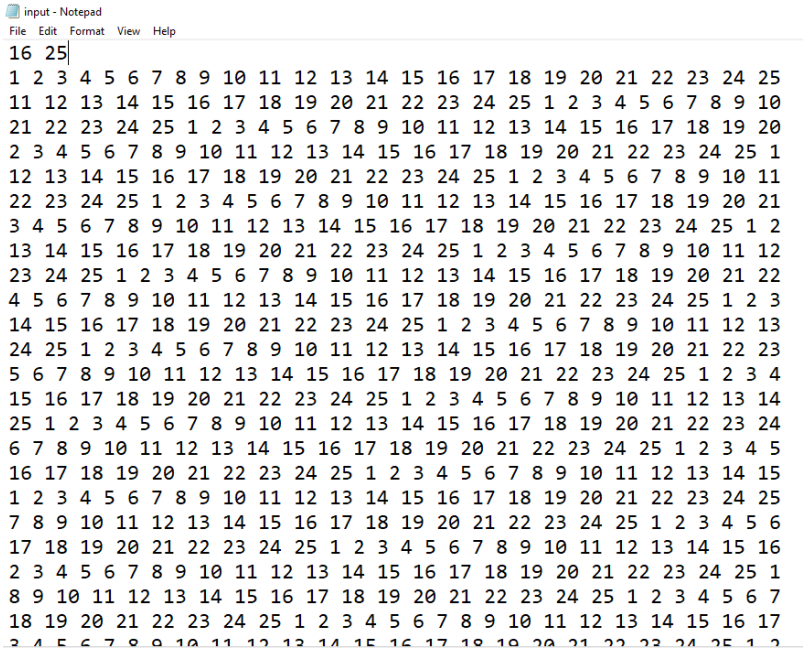
15 16 1 2 3 4 5 6 7 8 9 10 11 12 13 14

9 10 11 12 13 14 15 16 1 2 3 4 5 6 7 8

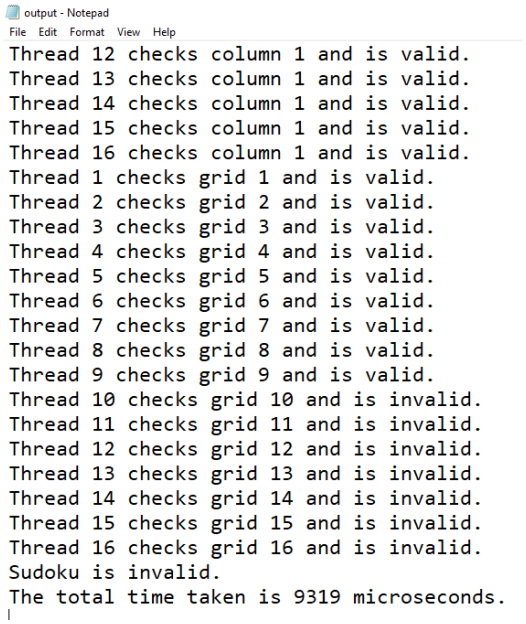
16*16 Output –



25*25 Input –



25*25 Output –



Similarly, we get the following outputs: -

OpenMP -

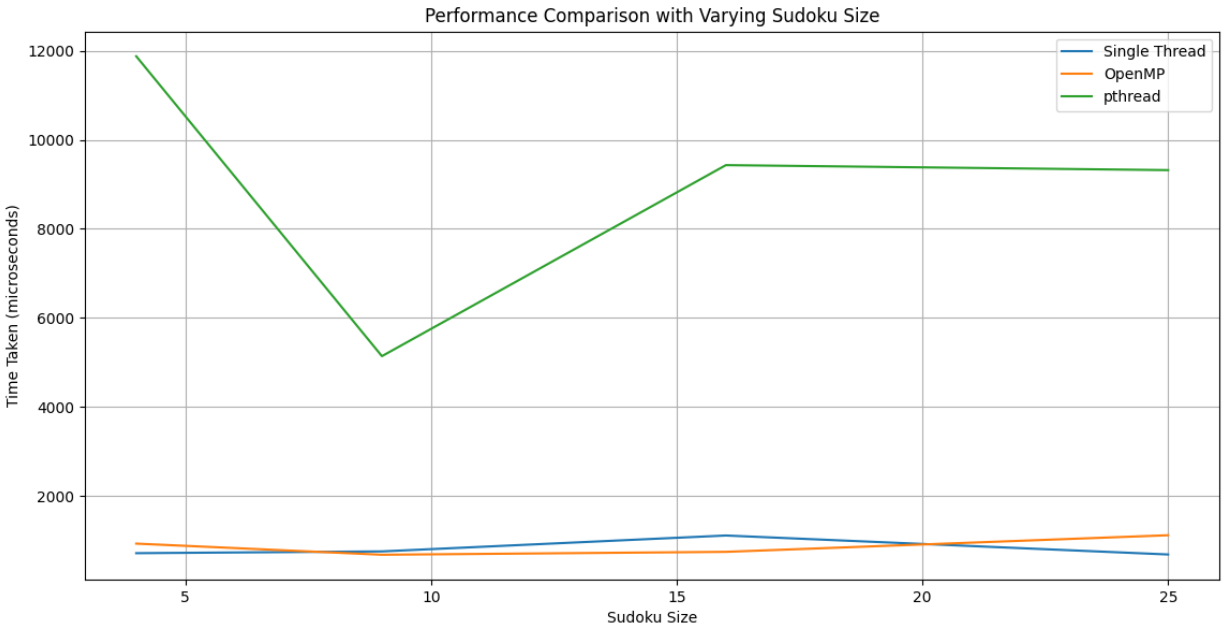
4*4 Output – 931 microseconds

9*9 Output – 681 microseconds

16*16 Output – 746 microseconds

25*25 Output – 1119 microseconds

Graph of Experiment 1:



Experiment 2 (Keeping threads changing, size constant**):**

Single Threading -

Output – 726 microseconds

Pthread -

2 Thread Output – 941 microseconds

4 Thread Output – 674 microseconds

8 Thread Output – 594 microseconds

16 Thread Output – 642 microseconds

OpenMP -

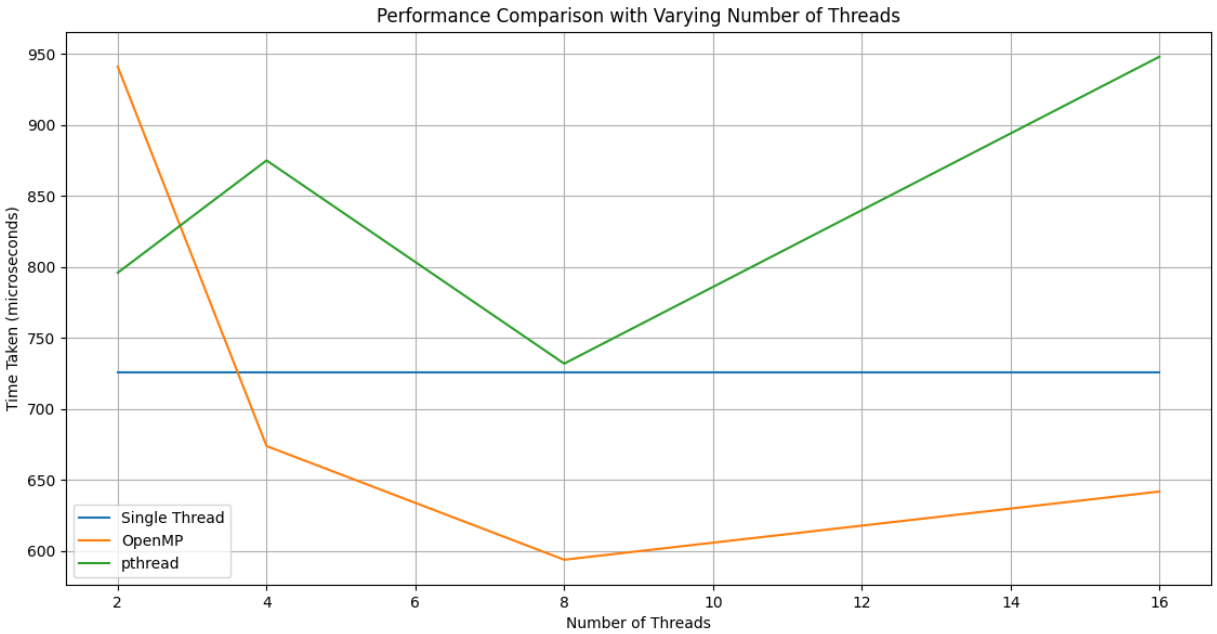
2 Thread Output – 796 microseconds

4 Thread Output – 875 microseconds

8 Thread Output – 732 microseconds

16 Thread Output – 948 microseconds

Graph of Experiment 2:



Conclusion

The results show that OpenMP performs better than Pthreads

Execution Time:

The OpenMP implementation generally shows shorter execution times compared to the Pthreads implementation across various sudoku sizes and thread counts.

Scalability:

OpenMP demonstrates superior scalability, with increasing thread counts, effectively parallelizing tasks and optimizing resource utilization compared to Pthreads.

Simplicity:

OpenMP presents a simpler and more straightforward parallel programming model in contrast to Pthreads.