

```
Обзор  Терминал ▾
[+]
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ g++ -fopenmp omp_lab_5.cpp -o omp_lab_5
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ ./omp_lab_5

OpenMP supported! Version 201511

Sum: 2082039698
Time: 0.0433762
Timer accuracy: 1e-09

ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ g++ -fopenmp omp_lab_5.cpp -o omp_lab_5
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ ./omp_lab_5

OpenMP supported! Version 201511

Sum: 2082039698
Time: 0.0433415
Timer accuracy: 1e-09
```

Случай одномерного массива. Сначала узнаем время выполнения нераспараллеленной программы, затем распараллеленной программы с помощью секции.

Как можно заметить разница во времени очень мала, но параллельная программа все же быстрее. Получить более существенную разницу на моем процессоре и моей оперативной памяти я не смог.

```
Обзор  Терминал ▾
[+]
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ g++ -fopenmp omp_lab_5.cpp -o omp_lab_5
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ ./omp_lab_5

OpenMP supported! Version 201511

Sum: 1498425601
Time: 0.0403657
Timer accuracy: 1e-09

ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ g++ -fopenmp omp_lab_5.cpp -o omp_lab_5
ortin-paler@mercury:~/5_semester/open_mp/lab/lab_5$ ./omp_lab_5

OpenMP supported! Version 201511

Sum: 1498425601
Time: 0.0433951
Timer accuracy: 1e-09
```

Случай двумерного массива. Сначала узнаем время выполнения нераспараллеленной программы, затем распараллеленной программы с помощью секции.

Ради интереса, я решил распараллелить еще и двумерный массив. В этот раз последовательные вычисления оказались все же быстрее, чем параллельные. Возможно существует гениальный способ поделить вычисления в двумерном массиве так, чтобы параллельный код работал быстрее при двух потоках, но я его придумать не смог(