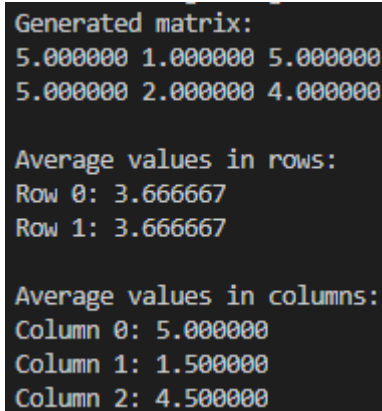


## Отчёт по лабораторной работе №4

Петрова О.А. группы ИВТ-12М

1. Разберите программу, представленную в файле `task_for_lecture5.cpp`. Предварительный просмотр документа. В программе создается 2 потока, каждый из которых вычисляет средние значения матрицы, один по строкам исходной матрицы `matrix`, а другой - по столбцам. Запустите программу и убедитесь в ее работоспособности.



```
Generated matrix:
5.000000 1.000000 5.000000
5.000000 2.000000 4.000000

Average values in rows:
Row 0: 3.666667
Row 1: 3.666667

Average values in columns:
Column 0: 5.000000
Column 1: 1.500000
Column 2: 4.500000
```

Рисунок 1. Результат программы `task_for_lecture5`

Результат вычисления верный.

2. Проанализируйте программу и введите в нее изменения, которые, по Вашему мнению, повысят ее производительность.

**Примечание:** в данной задаче (и последующих) обычные си-шные массивы с выделением и освобождением памяти заменены на вектора, т.к. у векторов не требуется задумываться о выделенной памяти.

Для повышения производительности добавим в функцию `FindAverageValues` параллелизм с помощью `cilk_for`, и редьюсер.

```
Generated matrix:  
4.000000 1.000000 3.000000  
5.000000 2.000000 3.000000  
  
Average values in rows:  
Row 0: 2.666667  
Row 1: 3.333333  
  
Average values in columns:  
Column 0: 4.500000  
Column 1: 1.500000  
Column 2: 3.000000
```

Рисунок 2. Результат вычисления средних значений после внесения вышеописанных изменений

Результат вычисления верный.

3. Определите с помощью Intel Parallel Inspector наличие в программе таких ошибок как: взаимная блокировка, гонка данных, утечка памяти. Сделайте скрины результатов анализа Parallel Inspector (вкладки Summary, Bottom-up) для всех упомянутых ошибок, где отображаются обнаруженные ошибки, либо отражается их отсутствие. Запускайте анализы на разных уровнях (Narrowest, Medium, Widest).

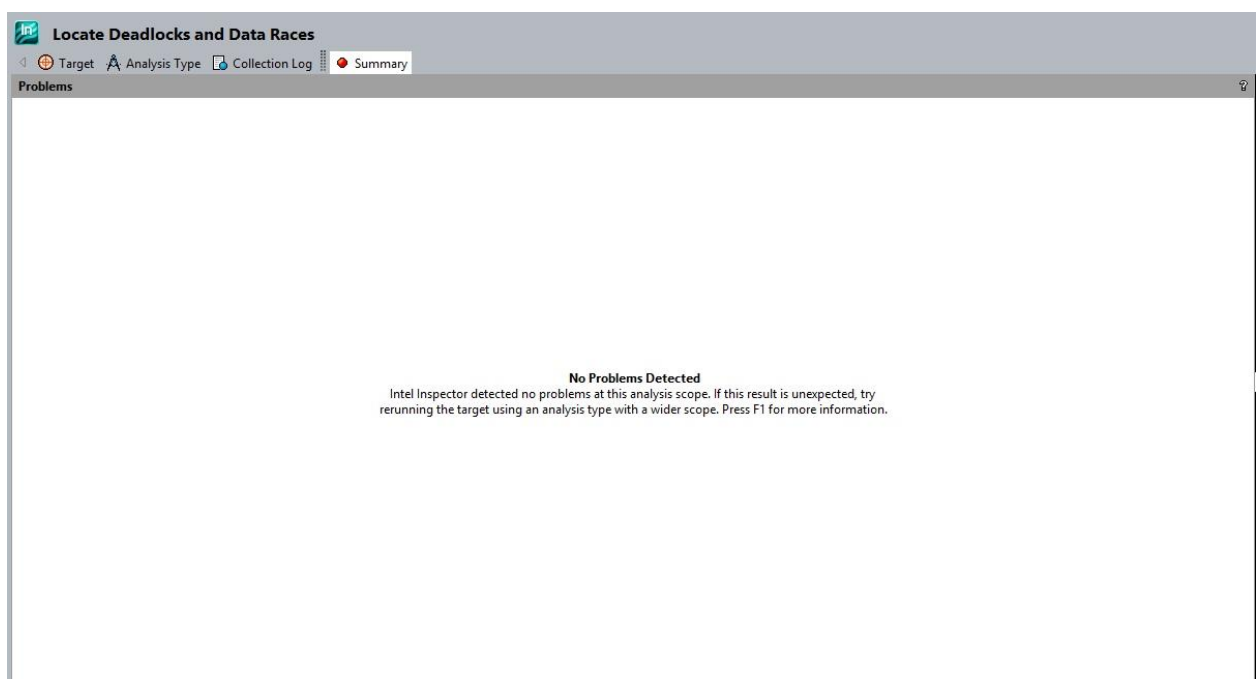


Рисунок 3. Результат анализа на выявление взаимных блокировок и гонки данных

В результате видно, что ошибок не обнаружено.

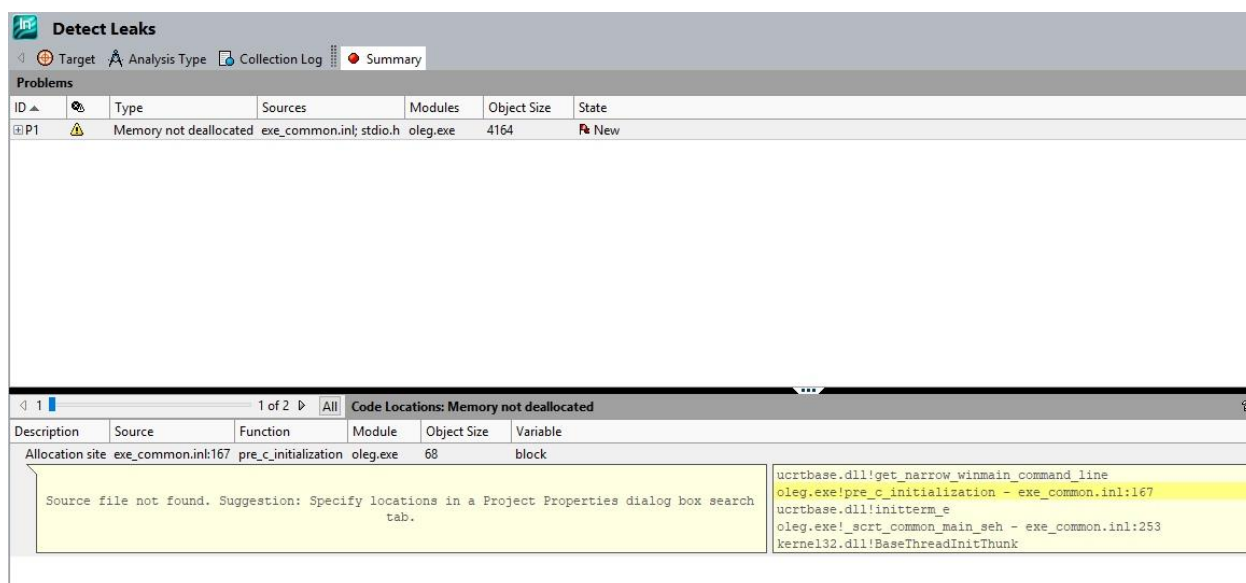


Рисунок 4. Результат анализа на выявление утечек памяти

В результате видно, что нашлось какое-то предупреждение что память не освобождена, но на форуме intel, где пользователь столкнулся с подобной ошибкой, предлагают проигнорировать это сообщение. В источнике указана стандартная библиотека и я понятия не имею что там могло пойти не так.

4. Измените код программы таким образом, чтобы Inspector при проверке не находил в программе ошибок, перечисленных в п. 3. Сделайте скрины результатов запуска Parallel Inspector.

Т.к. в задаче 3 не было обнаружено ошибок, то и изменений производить не нужно.

**Дополнение к задаче 3:** также были проведены анализы без `cilk_for` и `reducer`, и результаты анализов были точно такими же.