1. Разберите программу представленную в файле task_for_lecture5.cpp. 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:
1.000000 1.000000 4.000000
3.000000 3.000000 4.000000

Average values in rows:
Row 0: 2.000000
Row 1: 3.333333

Average values in columns:
Column 0: 2.000000
Column 1: 2.000000
Column 2: 4.000000
```

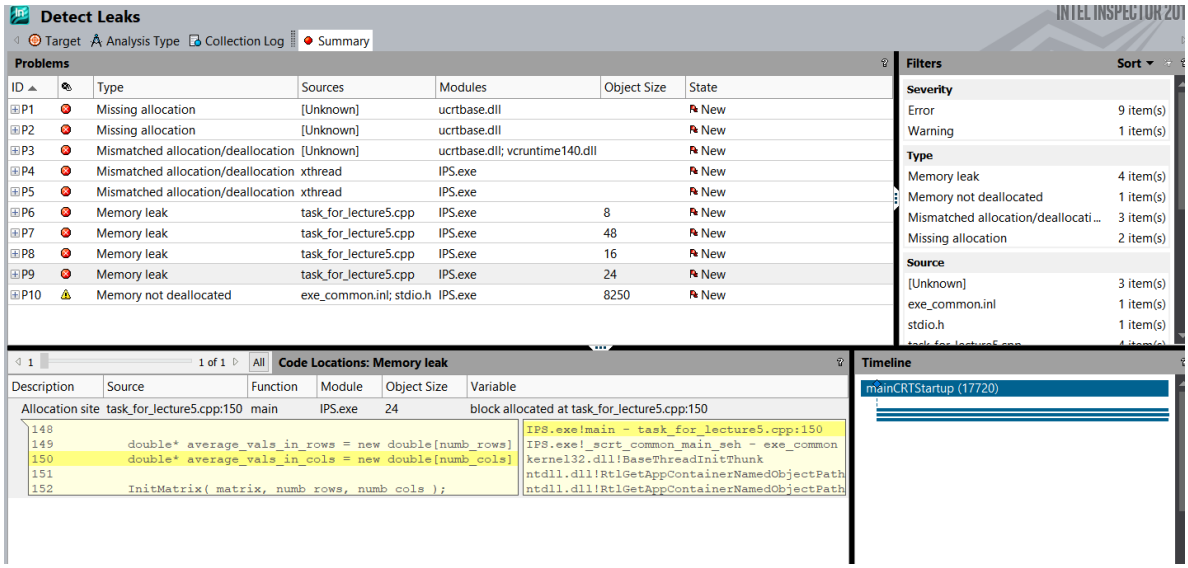
Программа работает правильно

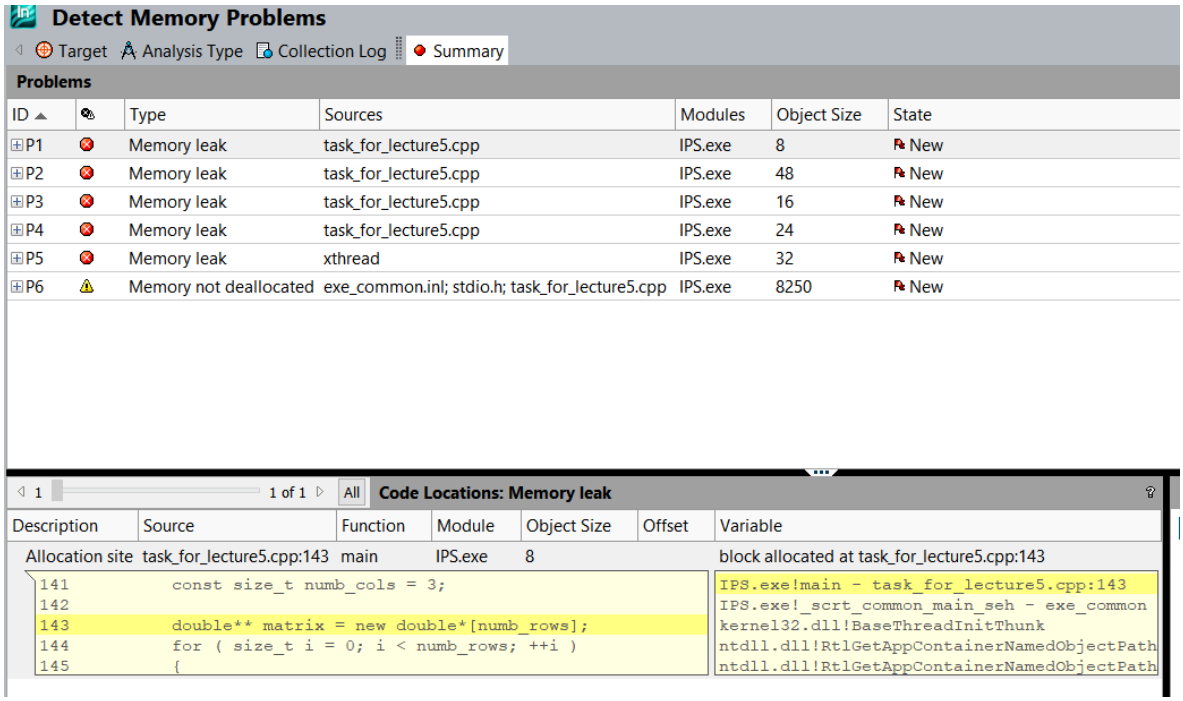
2. Проанализируйте программу и введите в нее изменения, которые по Вашему мнению повысят ее производительность.

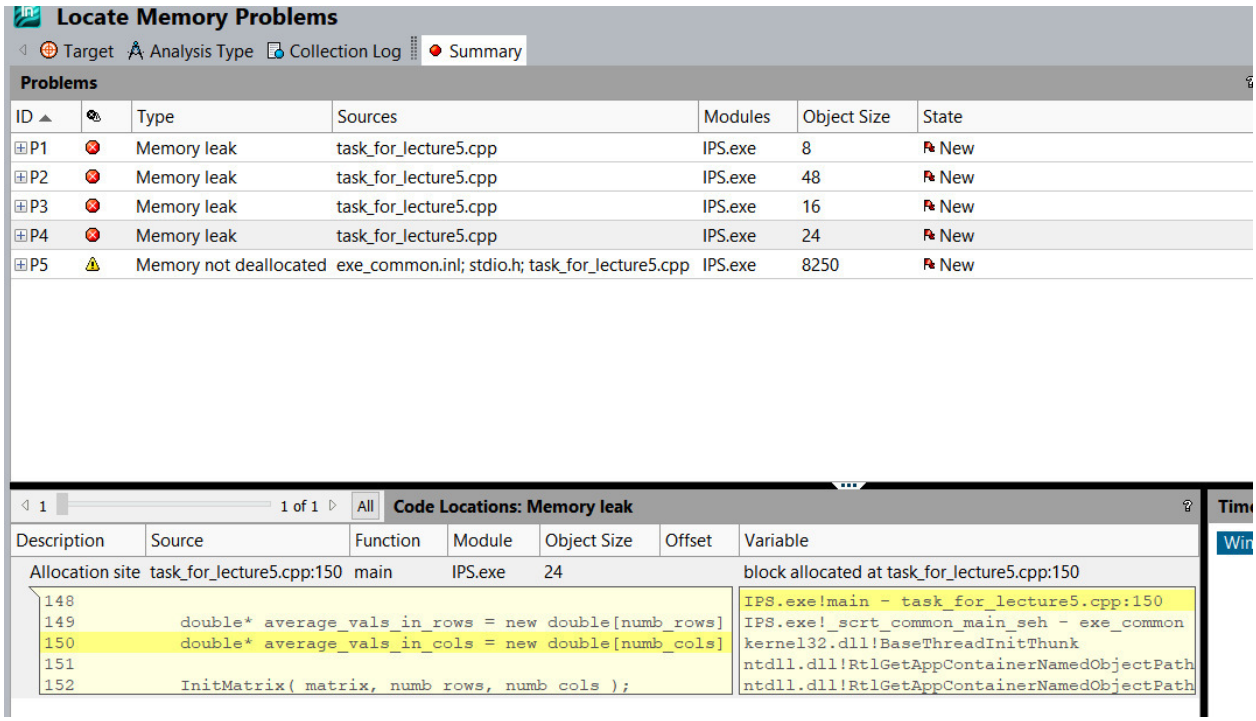
```
void FindAverageValues( eprocess_type proc_type, double** matrix, const size_t numb_rows, const size_t numb_cols, double* average_vals )
{
    switch ( proc_type )
    {
        case eprocess_type::by_rows:
        {
            cilk_for ( size_t i = 0; i < numb_rows; ++i )
            {
                //double sum( 0.0 );
                cilk::reducer_opadd<double> sum( 0.0 );
                cilk_for( size_t j = 0; j < numb_cols; ++j )
                {
                    sum += matrix[i][j];
                }
                //average_vals[i] = sum / numb_cols;
                average_vals[i] = sum.get_value() / numb_cols;
            }
            break;
        }
        case eprocess_type::by_cols:
        {
            cilk_for ( size_t j = 0; j < numb_cols; ++j )
            {
                //double sum( 0.0 );
                cilk::reducer_opadd<double> sum( 0.0 );
                cilk_for( size_t i = 0; i < numb_rows; ++i )
                {
                    sum += matrix[i][j];
                }
                //average_vals[j] = sum / numb_rows;
                average_vals[j] = sum.get_value() / numb_rows;
            }
            break;
        }
        default:
        {
            throw("Incorrect value for parameter 'proc_type' in function FindAverageValues() call!");
        }
    }
}
```

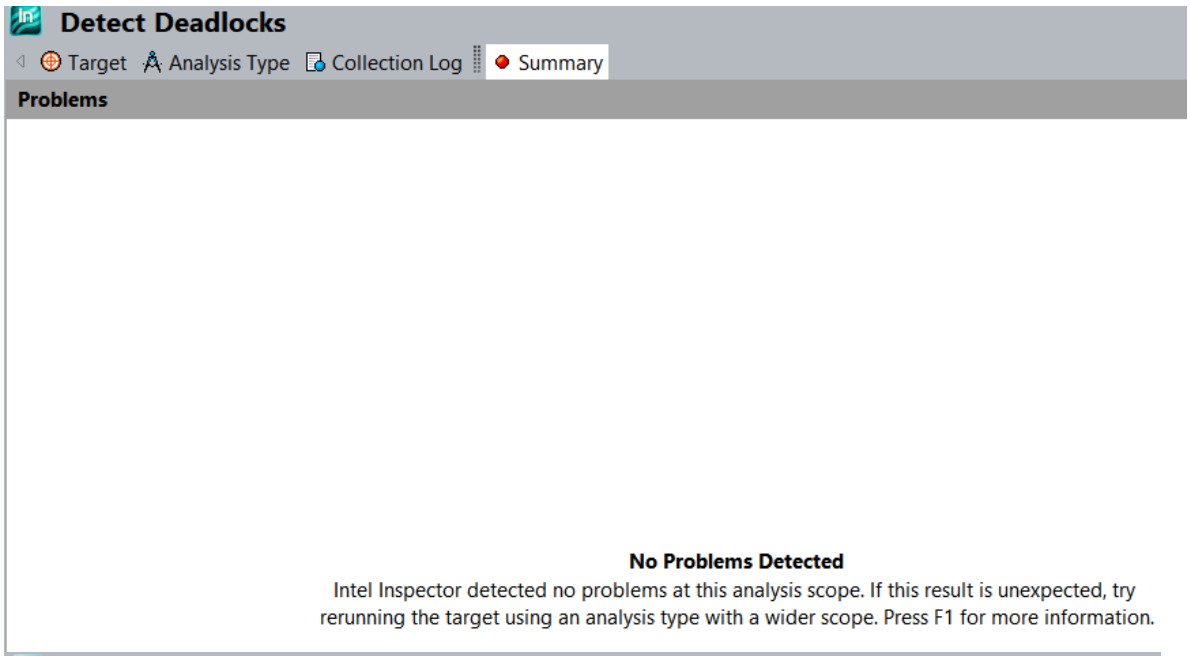
3. Определите с помощью ***Intel Parallel Inspector*** наличие в программе таких ошибок как: *взаимная блокировка*, *гонка данных*, *утечка памяти*. Сделайте скрины

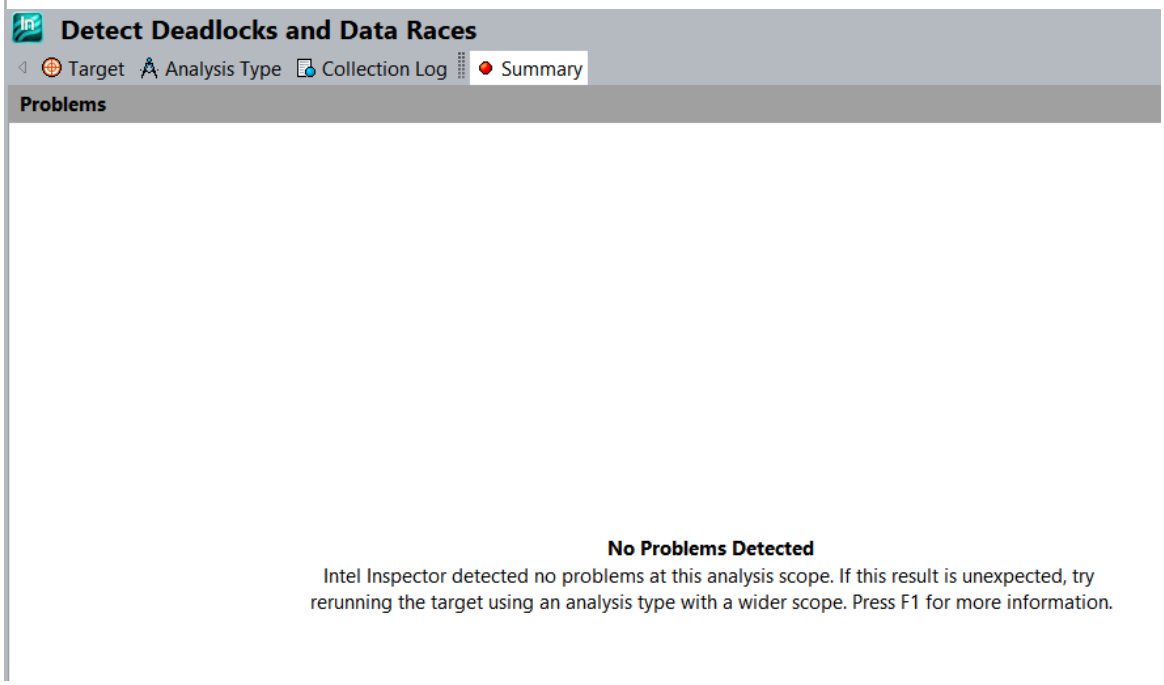
результатов анализа ***Parallel Inspector*** (вкладки ***Summary***, ***Bottom-up***) для всех упомянутых ошибок, где отображаются обнаруженные ошибки, либо отражается их отсутствие. Запускайте анализы на разных уровнях (***Narrowest***, ***Medium***, ***Widest***).

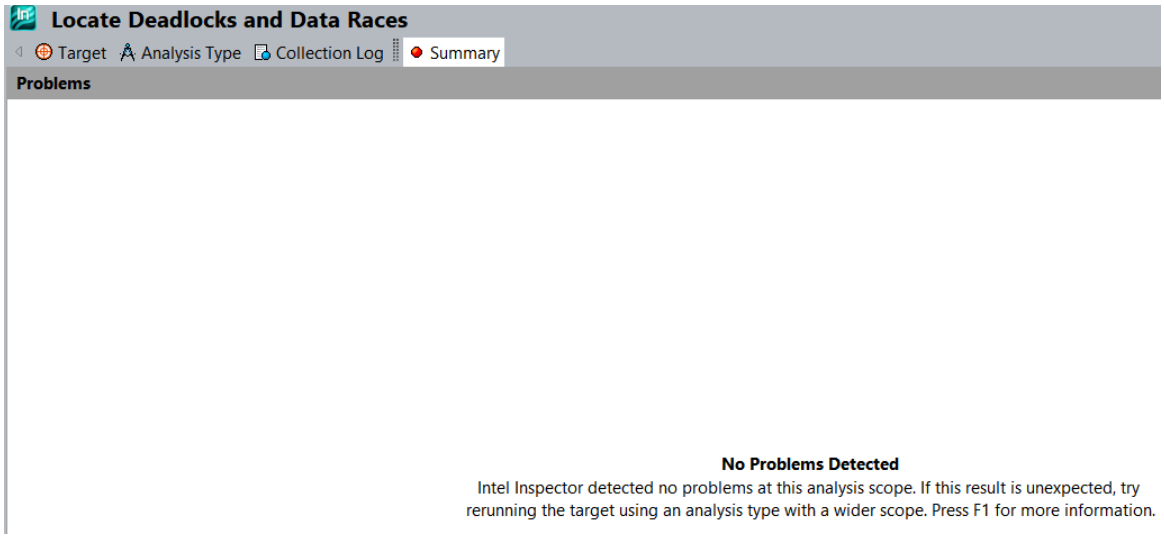





Locate Memory Problems
Target
Analysis Type
Collection Log
Summary
Problems
ID
Type
Sources
Modules
Object Size
State
P1
Memory leak
task_for_lecture5.cpp
IPS.exe
8
New
P2
Memory leak
task_for_lecture5.cpp
IPS.exe
48
New
P3
Memory leak
task_for_lecture5.cpp
IPS.exe
16
New
P4
Memory leak
task_for_lecture5.cpp
IPS.exe
24
New
P5
Memory not deallocated
exe_common.inl; stdio.h; task_for_lecture5.cpp
IPS.exe
8250
New
1
1 of 1
All
Code Locations: Memory leak
Description
Source
Function
Module
Object Size
Offset
Variable
Allocation site
task_for_lecture5.cpp:150
main
IPS.exe
24
block allocated at task_for_lecture5.cpp:150
148
149
double* average_vals_in_rows = new double[numb_rows]
150
double* average_vals_in_cols = new double[numb_cols]
151
152
InitMatrix(matrix, numb_rows, numb_cols);
IPS.exe!main - task_for_lecture5.cpp:150
IPS.exe!_scrt_common_main_seh - exe_common
kernel32.dll!BaseThreadInitThunk
ntdll.dll!RtlGetAppContainerNamedObjectPath
ntdll.dll!RtlGetAppContainerNamedObjectPath


Detect Deadlocks
Target
Analysis Type
Collection Log
Summary
Problems
No Problems Detected
Intel Inspector detected no problems at this analysis scope. If this result is unexpected, try rerunning the target using an analysis type with a wider scope. Press F1 for more information.

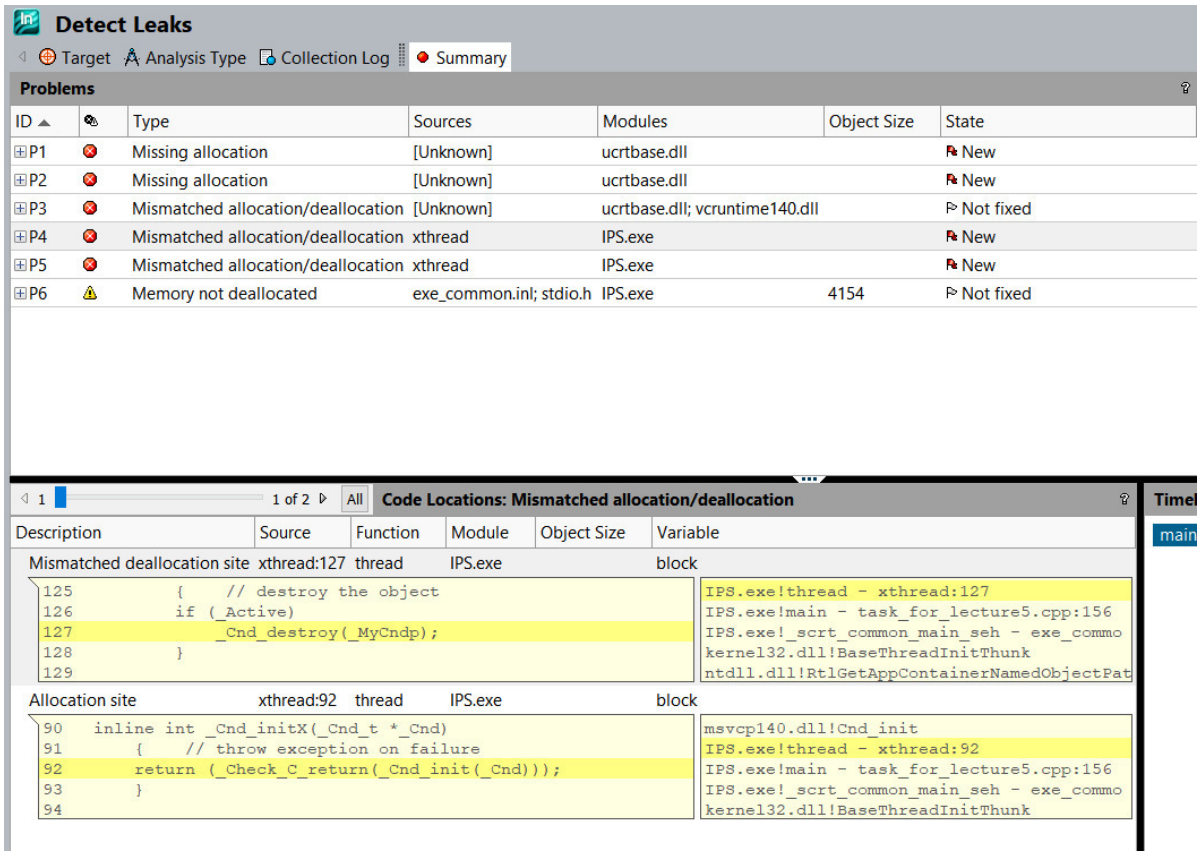

Detect Deadlocks and Data Races
Target
Analysis Type
Collection Log
Summary
Problems
No Problems Detected
Intel Inspector detected no problems at this analysis scope. If this result is unexpected, try rerunning the target using an analysis type with a wider scope. Press F1 for more information.

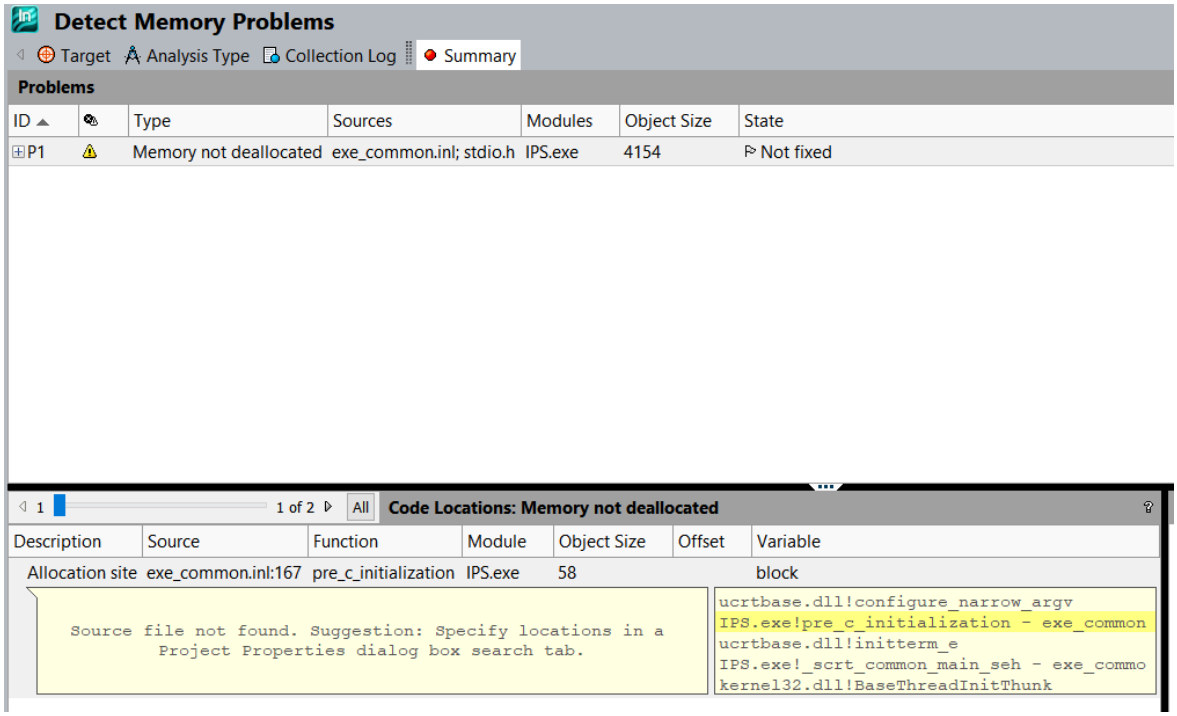

Locate Deadlocks and Data Races
Target
Analysis Type
Collection Log
Summary
Problems
No Problems Detected
Intel Inspector detected no problems at this analysis scope. If this result is unexpected, try rerunning the target using an analysis type with a wider scope. Press F1 for more information.

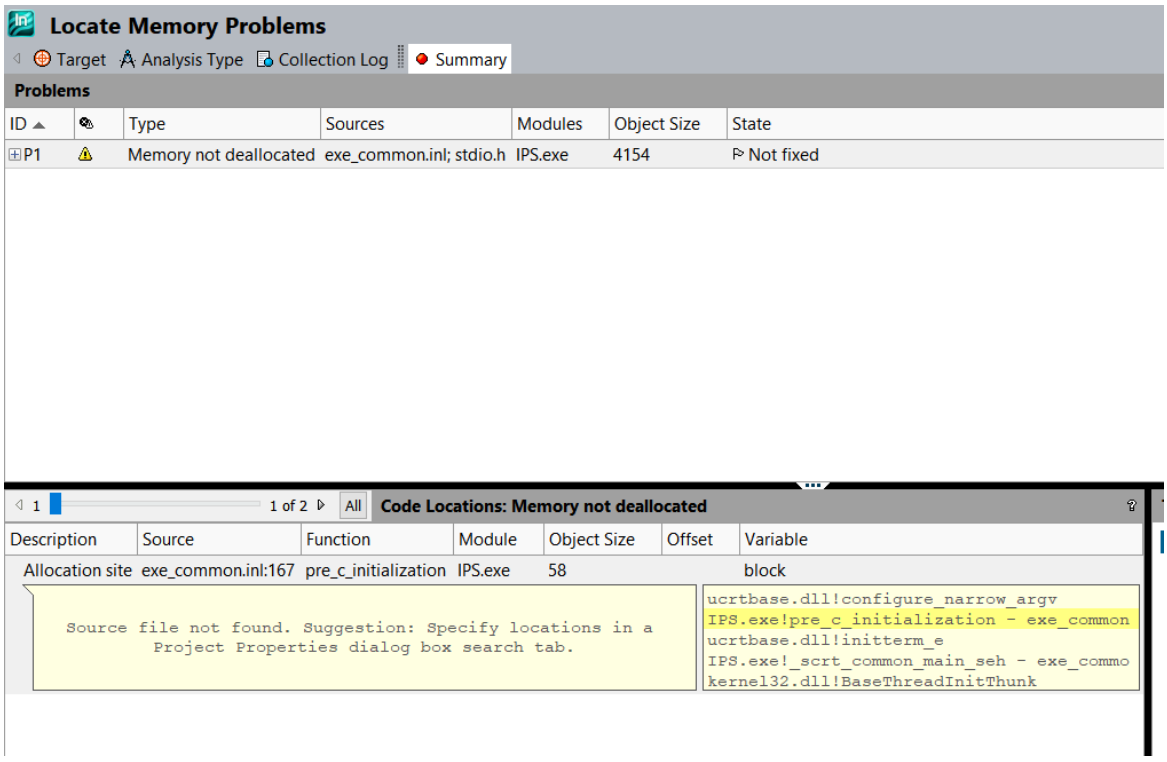
4. Измените код программы таким образом, чтобы ***Inspector*** при проверке не находил в программе ошибок, перечисленных в **п. 3.** Сделайте скрины результатов запуска ***Parallel Inspector***.

```
for (size_t i = 0; i < numb_rows; i++)
{
    delete[] matrix[i];
}
delete[] matrix;
delete[] average_vals_in_rows;
delete[] average_vals_in_cols;
```

Код добавлен в конец блока try функции main().







Как можно видеть, ошибки, связанные с утечкой памяти в коде файла task_for_lecture5.cpp, исправлены