**Измерение производительности приложения посредством анализа использования ЦП**

Цели нашей работы – Сбор данных об использовании ЦП и Анализ данных о загрузке ЦП. Проверка данной программы на другом ПО

**Шаг 1. Сбор данных профилирования**

1. Запустили проект в программе Visual Studio и разместили две точки остановы где хотим проверить нагрузку на ЦП и начали отладку программы проверку программы.

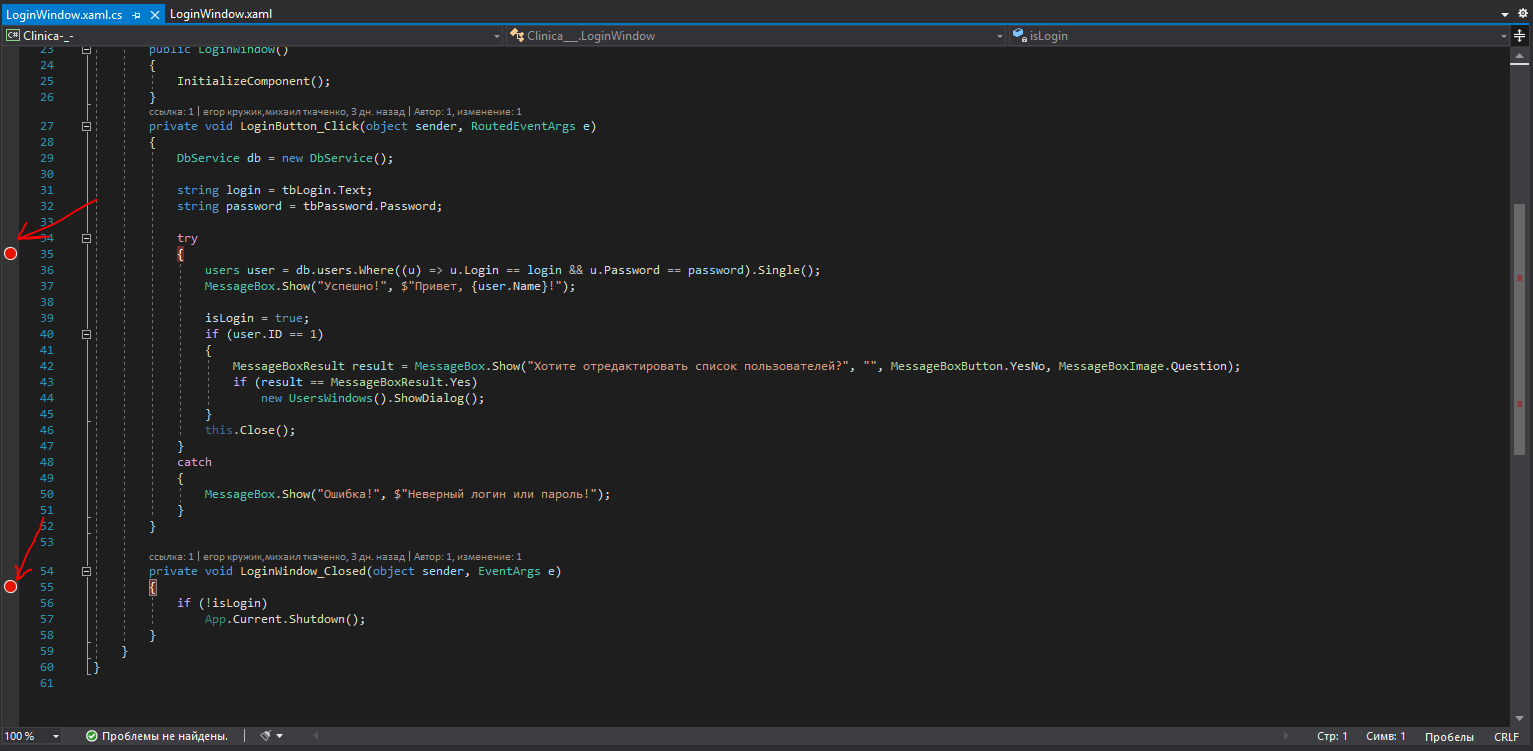


Рис. 1. Код с точками остановы

2. Ставим галочку на кнопке «Запись профиля ЦП”. После этого, программа дойдёт до второй точки остановы, ждём пока закончится обработка данных.

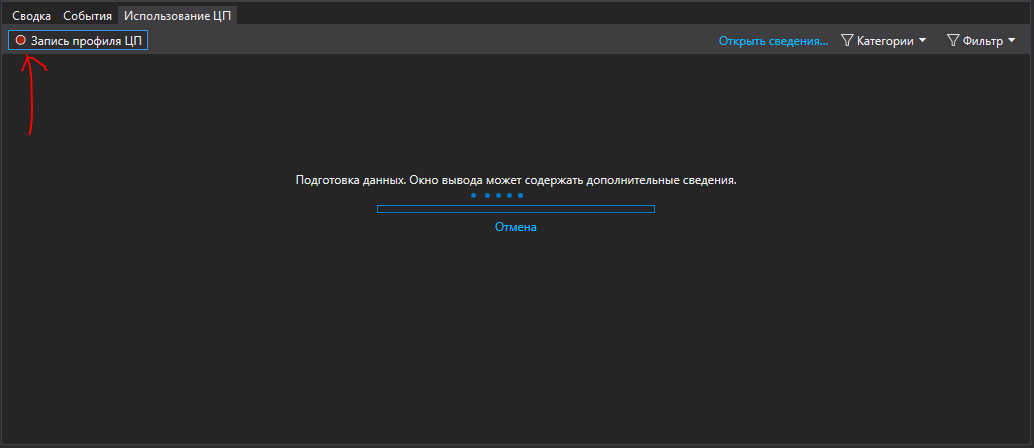


Рис. 2. Использование ЦП

3. После того как обработка данных завершена, мы видим, все данные по загрузке ЦП

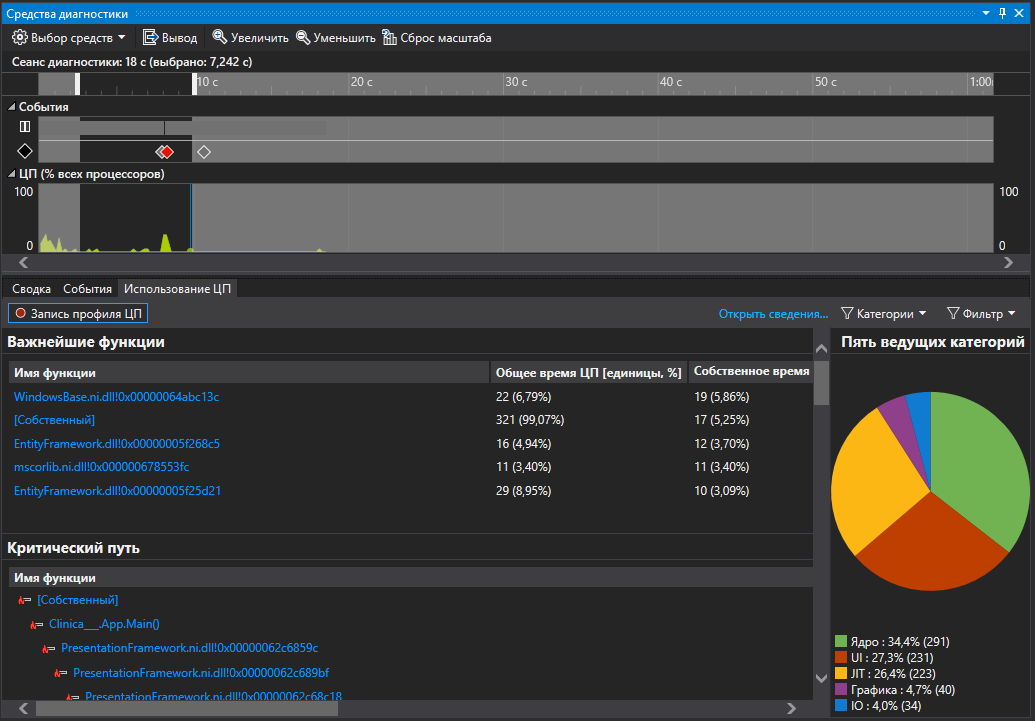


Рис. 3. Результаты загрузки ЦП

**Шаг 2. Анализ данных о загрузке ЦП**

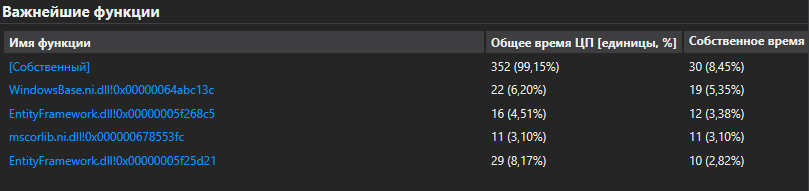
4. На этом скрине мы видим список функций, которые выполняют большую часть работы. После нажатия на одну из функций в левой панели откроется представление **Вызывающий/вызываемый**.

Рис. 4. Важнейшие функции

5. В списке функций изучите функции, которые выполняют большую часть работы

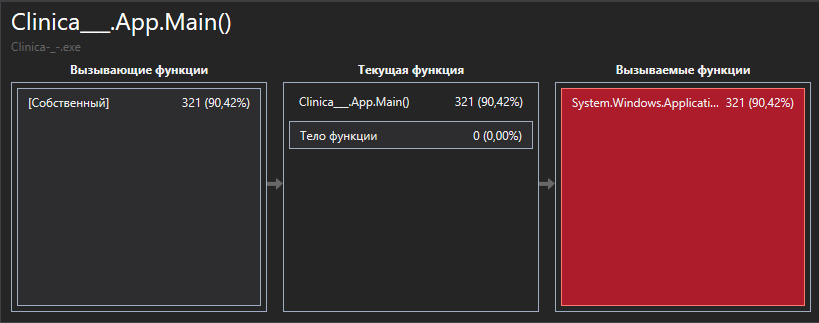


Рис. 5. Представление «Вызывающий/вызываемый»

6. Чтобы увидеть более обобщенное представление, показывающее порядок, в котором вызываются функции, выберите в раскрывающемся списке в верхней части панели пункт **Дерево вызовов**

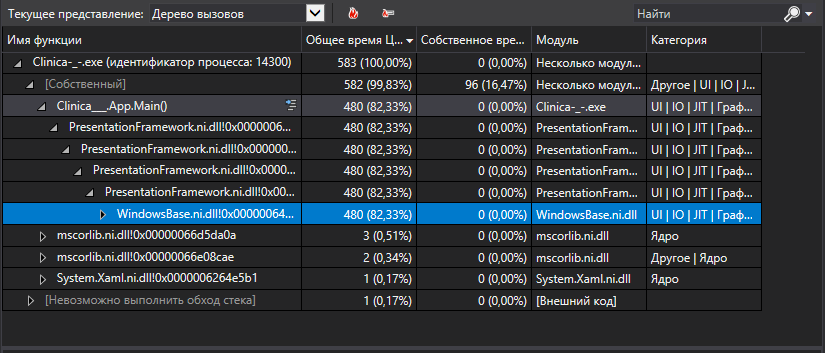


Рис. 6. Представление «Древо вызовов»

7. Далее нажимаем кнопку «Показать горячий путь»

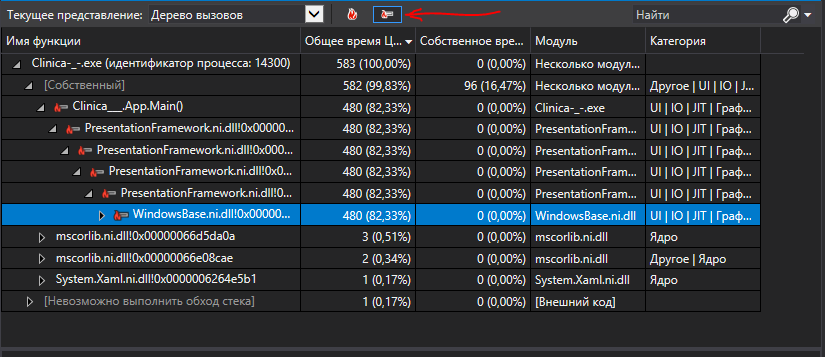


Рис. 7. Представление «Древо вызовов» с включённой функцией «Показать горячий путь»

8. Далее идёт функция «Просмотр внешнего кода»: для этого ставим галочку на нужной кнопке

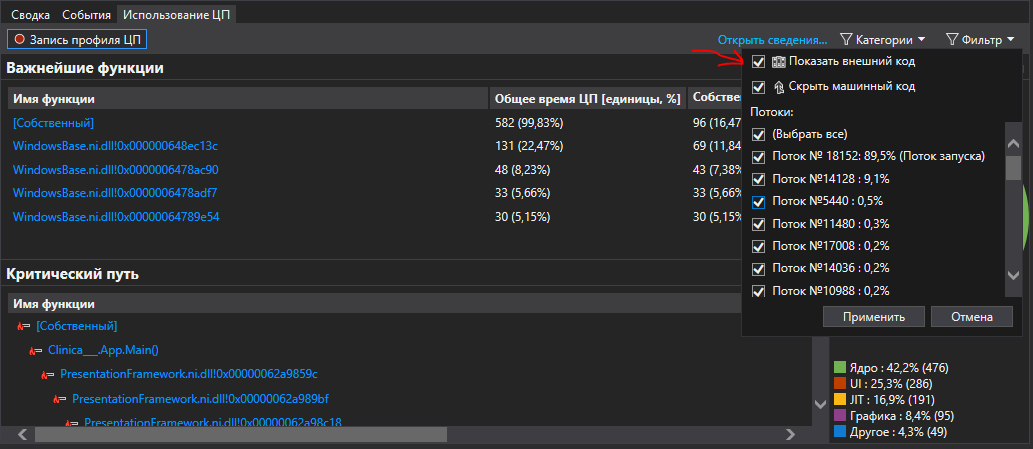


Рис. 8. Функция «Показать внешний код»

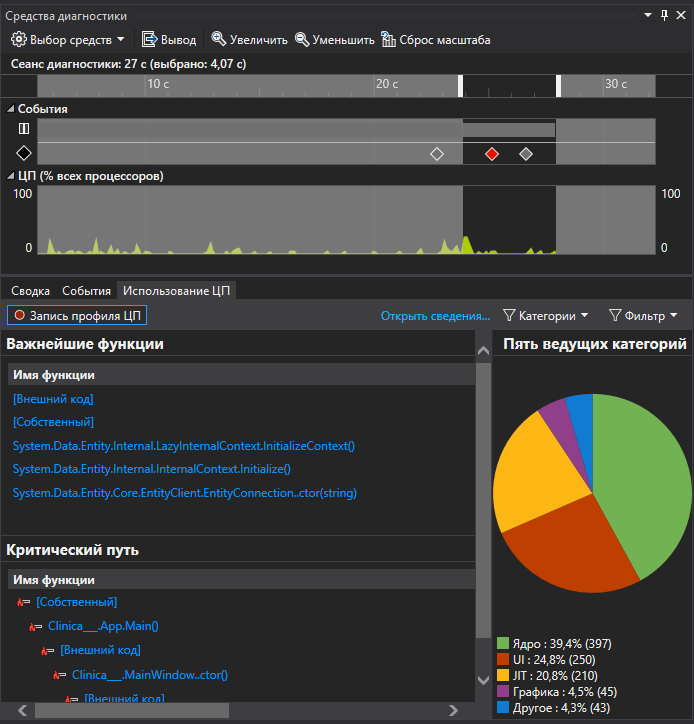
Данные по загрузке ЦП на другом ПК.

Рис. 9. Проверка на другом ПК