Задание на практику №1

**Измерение производительности приложения посредством анализа использования ЦП**

1. Выбрать Отладка 🡪 Окно 🡪 Показать средства диагностики

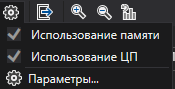


Рисунок 1 – «Окно»

1. Нажать Отладка 🡪 Начать отладку (F5) и зайти в средства диагностики

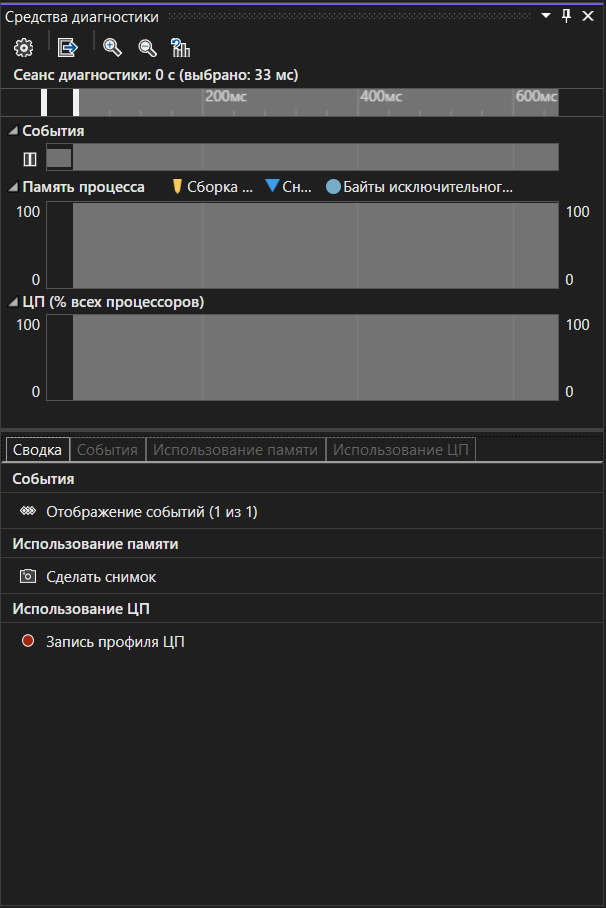


Рисунок 2 – «Средства диагностики»

1. Приостановив отладчик, включить сбор данных о загрузке и открыть вкладку **Загрузка ЦП**

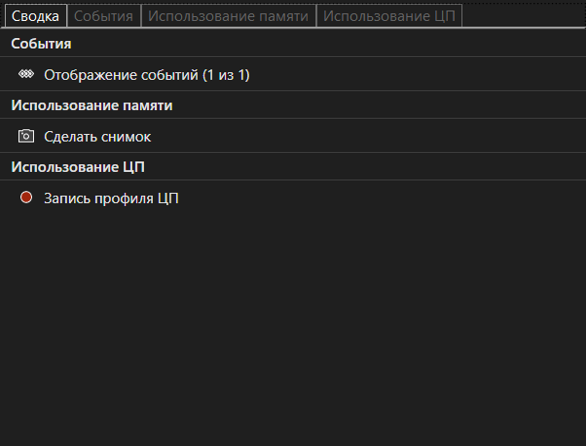


Рисунок 3 – «Сводка»

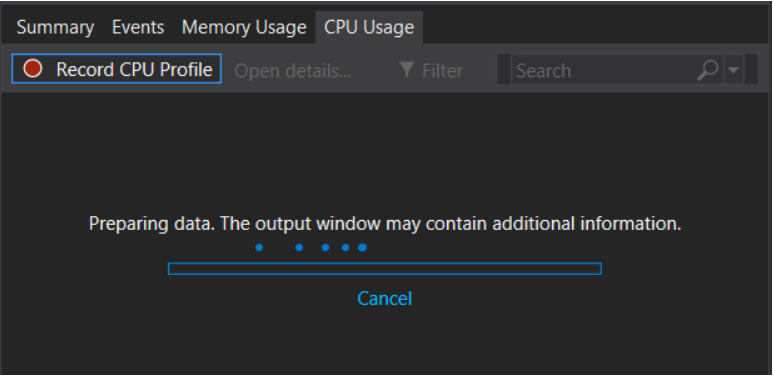


Рисунок 4 – «Загрузка ЦП»

1. После запуска откладки запустится приложение до второй точки останова, средство «Загрузка ЦП» выведет отчет на вкладке **Загрузка ЦП**

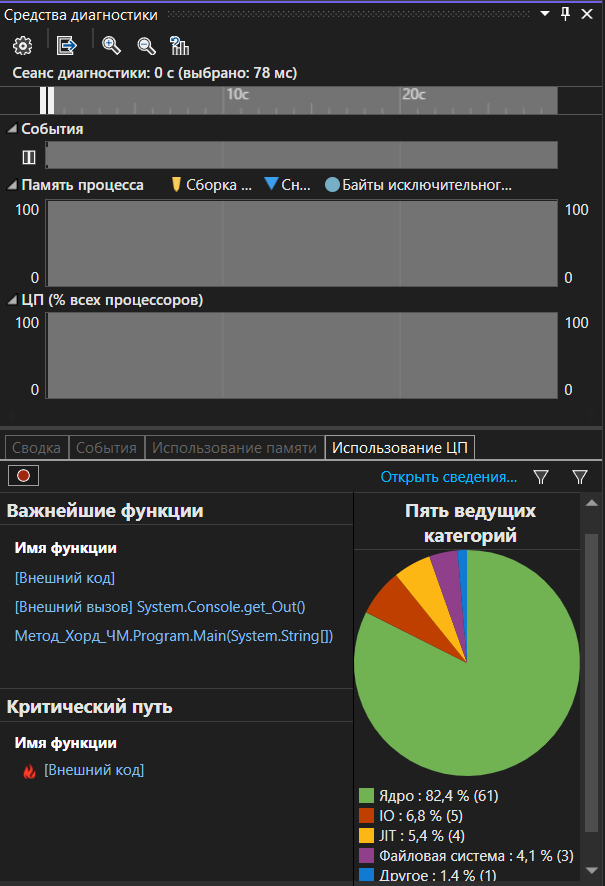


Рисунок 5 – «Отчет»

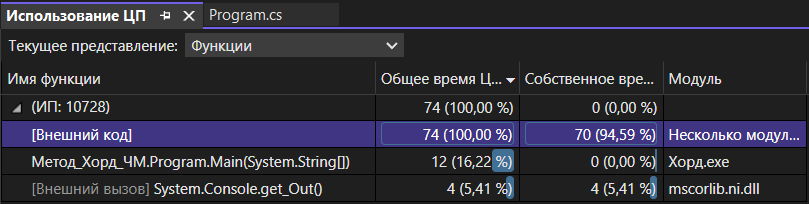


Рисунок 6 – «Использование ЦП»

1. Конкретная выбранная область на временной шкале



Рисунок 7 – «Временная шкала»

1. При двойном щелчке функции в левой панели откроется представление **Вызывающий/вызываемый**

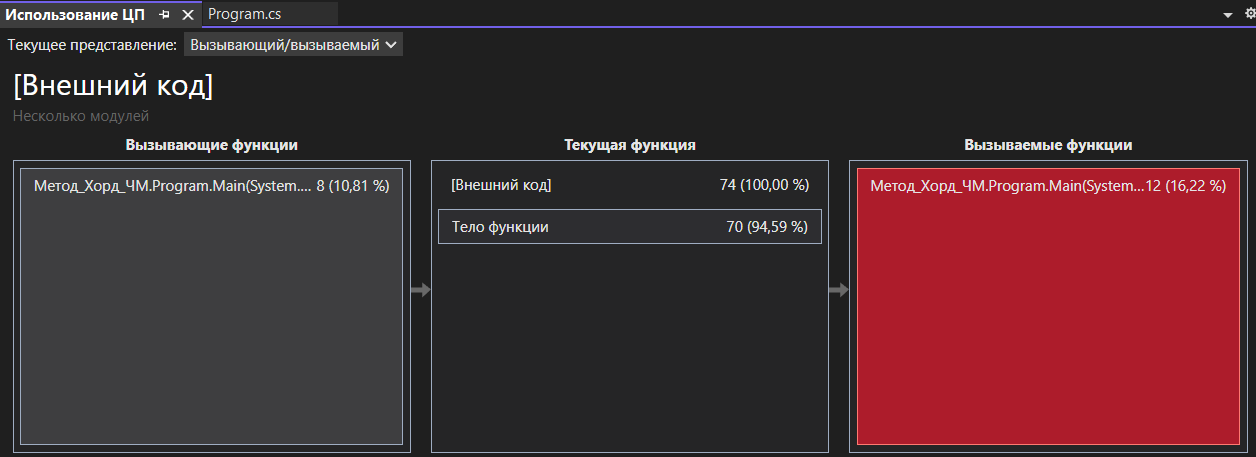


Рисунок 8 – «Вызывающий/вызываемый»

1. Изменить параметр представления на **Дерево вызовов**, чтобы увидеть более обобщенный вид

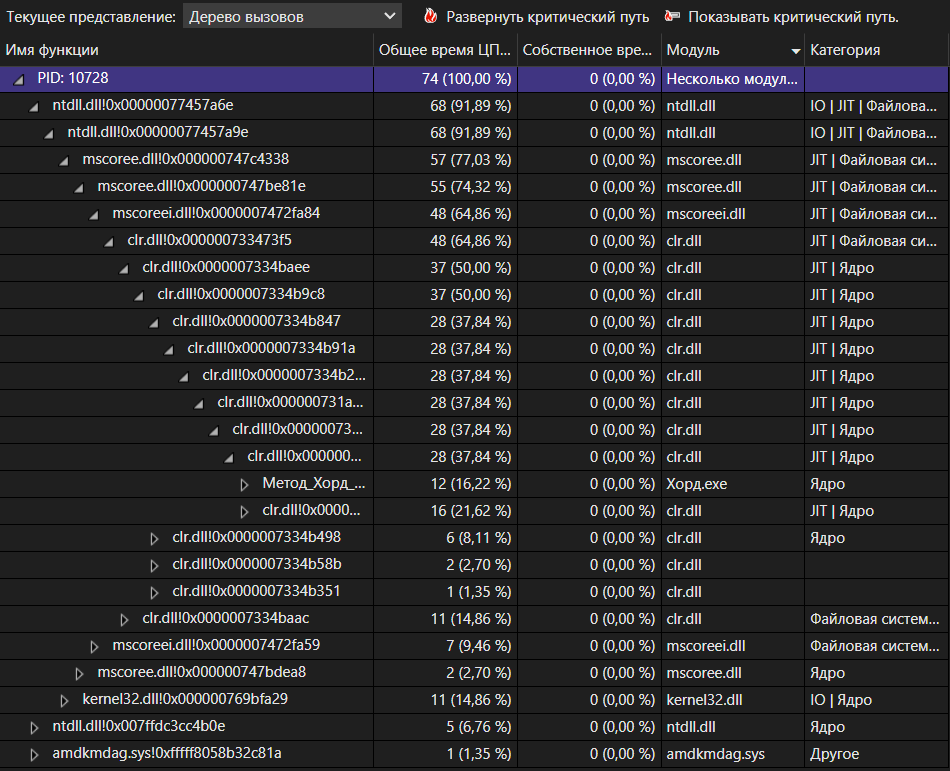


Рисунок 9 – «Дерево вызовов»

1. Нажать на параметр **Развернуть критический путь**, чтобы увидеть вызовы функций, которые используют самый высокий процент ЦП в представлении дерева вызовов

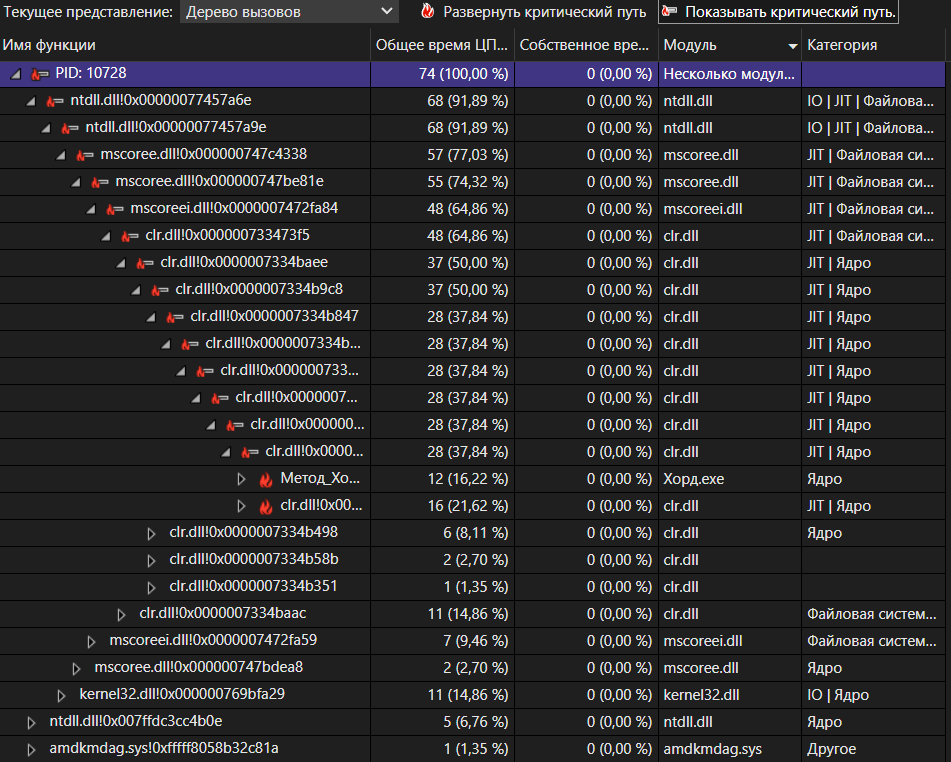


Рисунок 10 – «Развернуть критический путь»

1. Нажать **Показать внешний код** в списке **Представление фильтра** и выберите **Применить**

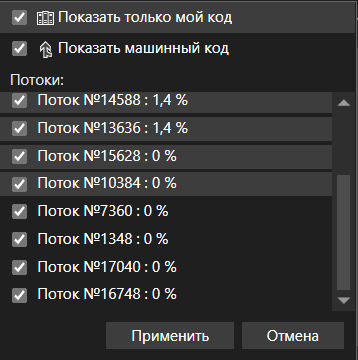


Рисунок 11 – «Показать внешний путь»

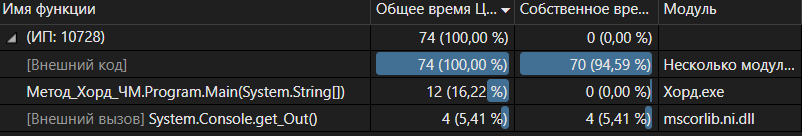


Рисунок 12 – «Функции»

1. Отправка завершенной работы на GitHub

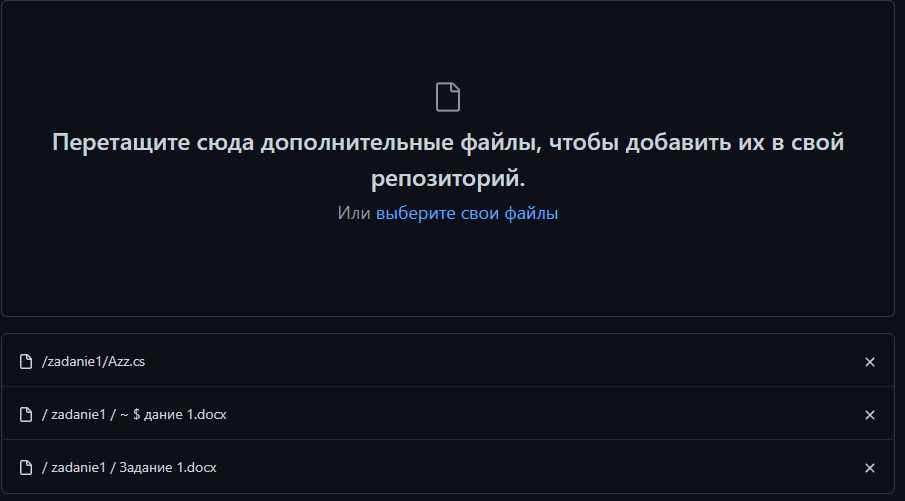


Рисунок 13 – «Отправка работы»