Измерение производительности приложения посредствам анализа использования ЦП

1. Запускаем Visual Studio
2. Открываем проект, который нужно проанализировать.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 1 “Окно открытия проекта”

1. Устанавливаем первую точку остановки в программе.

**Изображение выглядит как текст

Автоматически созданное описание**

Рис. 2 “Первая точка остановки”

1. Устанавливаем вторую точку остановки в конце функции.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 3 “Вторая точка остановки”

1. Открываем окно *Отладка>Окна>Показать средства диагностики*

Изображение выглядит как текст, снимок экрана, монитор, черный

Автоматически созданное описание

Рис. 4 “Средство диагностики”

1. Выбираем *Использование памяти* и *Загрузку ЦП*. Так же можем включить или выключить IntelliTrace, выбрав *Сервис> Параметры > IntelliTrace*.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 4 “Параметры”

1. Выбираем *Отладка> Начать отладку* или (*Запустить* на панели инструментов или *F5*).

Изображение выглядит как текст

Автоматически созданное описание

Рис. 7 “Окно отладки”

1. Запускаем сценарий, который вызвал срабатывание первой точки остановки в программе.

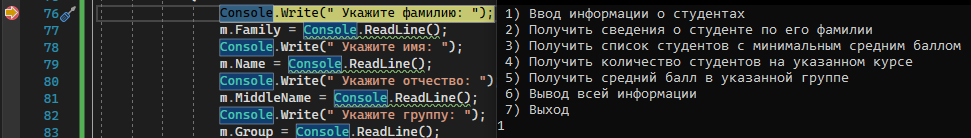
**

Рис. 8 “Срабатывание первой точки остановки”

1. Приостановив отладчик, включаем сбор данных об загрузке ЦП и откроем закладку *Загрузка ЦП*.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 9 “Начало сбора данных об загрузке ЦП”

1. Продолжаем работу отладчика с помощью клавиши *F5*, чтоб запустить программу до следующей точке остановки.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 10 “Срабатывание второй точки остановки”

Теперь у нас есть данные о производительности программы, именно в той области кода, которая выполняется между двумя точками остановки.

Изображение выглядит как текст, монитор, снимок экрана, табло

Автоматически созданное описание

Рис. 11 “Данные о загрузке ЦП”

1. Для более точного анализа можно выбрать более конкретную область кода, выбрав на временной шкале ЦП.

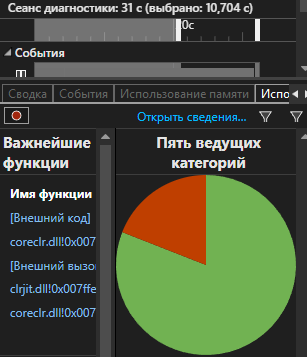


Рис. 12 “Загрузка ЦП на определённом отрезке времени”

Шаг 2. Анализ данных о загрузке ЦП

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

Изображение выглядит как текст, снимок экрана, монитор, внутренний

Автоматически созданное описание

Рис. 13 “Вызываемые функции”

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

Изображение выглядит как текст, снимок экрана, экран

Автоматически созданное описание

Рис. 14 “Дерево вызовов”

Чтоб увидеть вызовы функций, которые используют самый высокий процент ЦП в представлении древа вызовов, нажмите *Развернуть критический пусть*.

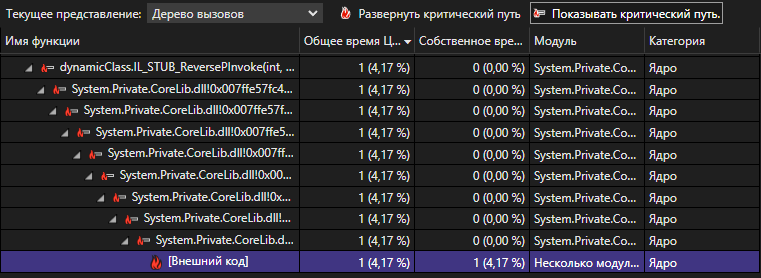


Рис. 15 “Критический путь”

1. Просмотрим внешний код, для этого выберем в раскрывающем списке *Функции*.

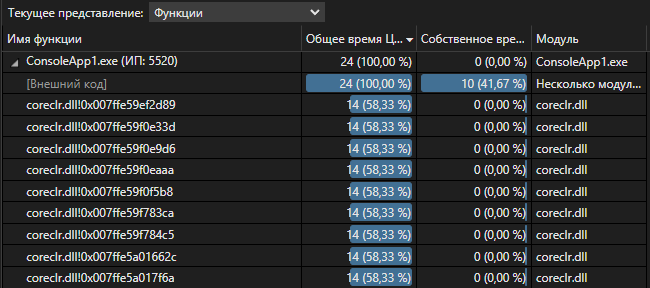


Рис. 16 “Функции”