**Отчет по практике**

1. Откройте проект для отладки в Visual Studio и установите точку останова в приложении в точке, где вы хотите проверить загрузку ЦП.
2. Установите вторую точку останова в конце функции или области кода, который требуется проанализировать.

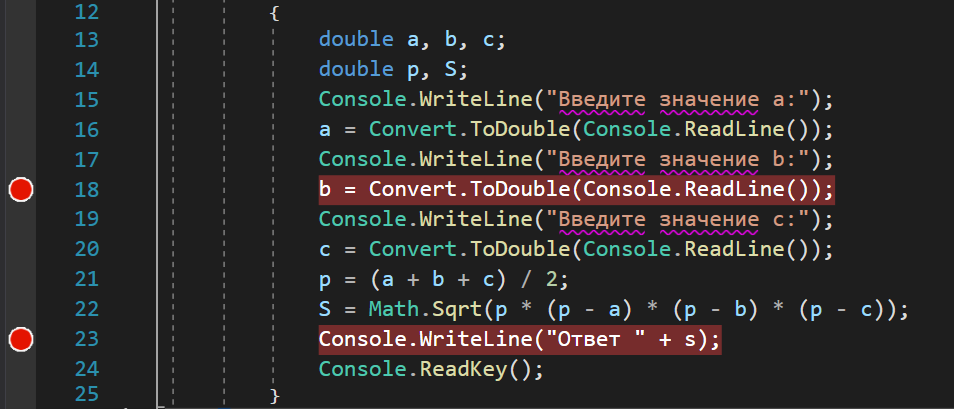


Рис.1

3. Окно **Средства диагностики** появится автоматически, если вы не отключали эту функцию. Чтобы снова открыть окно, щелкните **Отладка** > **Окна** > **Показать средства диагностики**.

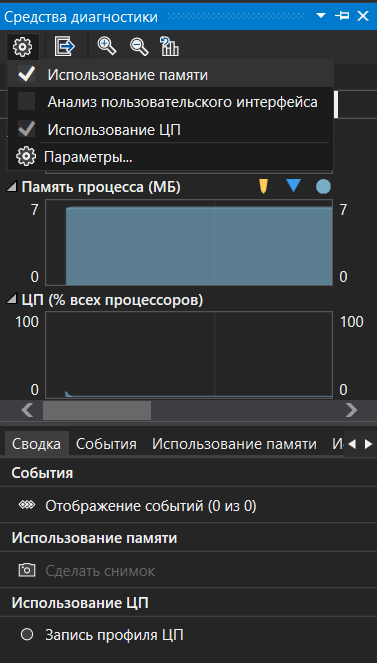


Рис.2

4. Вы можете выбрать, что следует просмотреть, [Использование памяти](https://docs.microsoft.com/ru-ru/visualstudio/profiling/memory-usage?view=vs-2022) или **Загрузка ЦП** (либо оба средства), с помощью параметра **Выбор средств** на панели инструментов. В Visual Studio Enterprise также можно включить или отключить IntelliTrace, выбрав **Сервис** > **Параметры** > **IntelliTrace**.

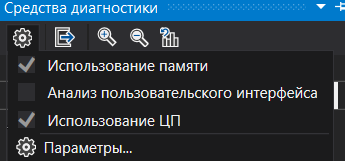


Рис.3

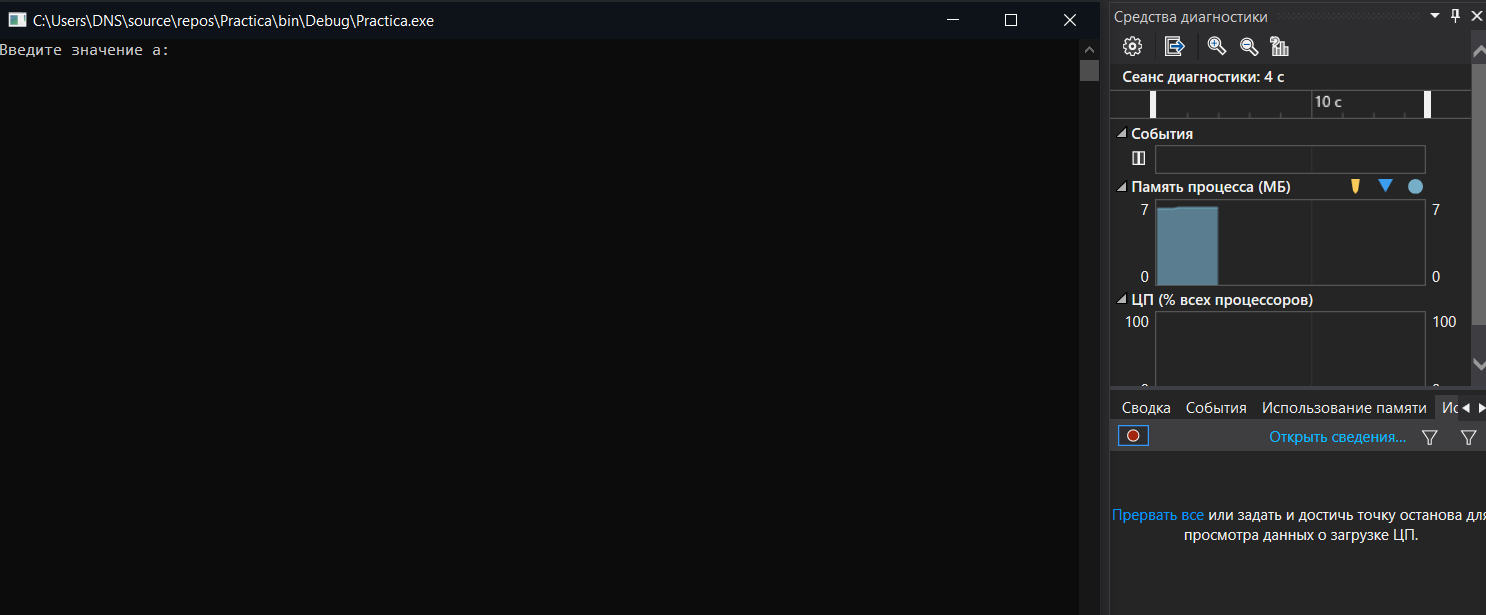
5. Щелкните **Отладка** > **Начать отладку** (**Запустить** на панели инструментов или **F5**).

Рис.4

6. Запустите сценарий, который вызвал срабатывание первой точки останова.

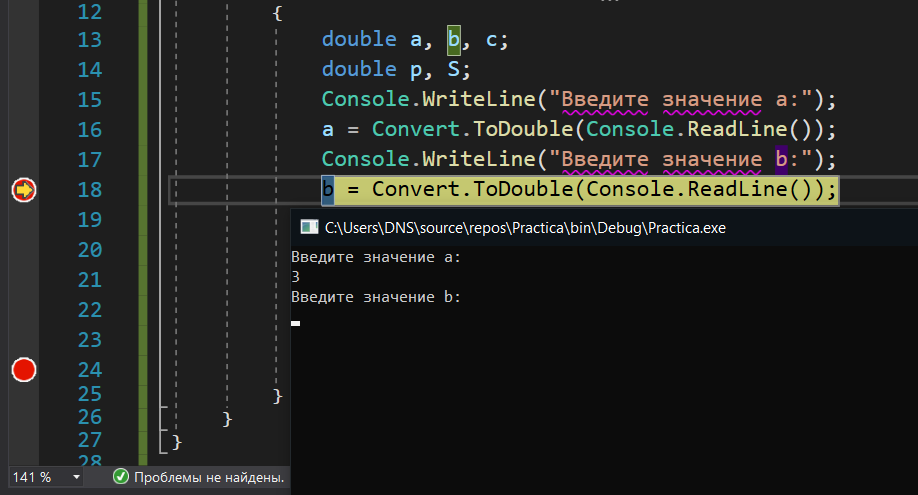


Рис.5

7. Приостановив отладчик, включите сбор данных о загрузке ЦП, а затем откройте вкладку **Загрузка ЦП**.



Рис.6

8. Нажмите клавишу F5, чтобы запустить приложение до второй точки останова.

Теперь у вас есть данные о производительности приложения именно для той области кода, которая выполняется между двумя точками останова.

Профилировщик начинает подготавливать данные потока. Дождитесь завершения этой операции.

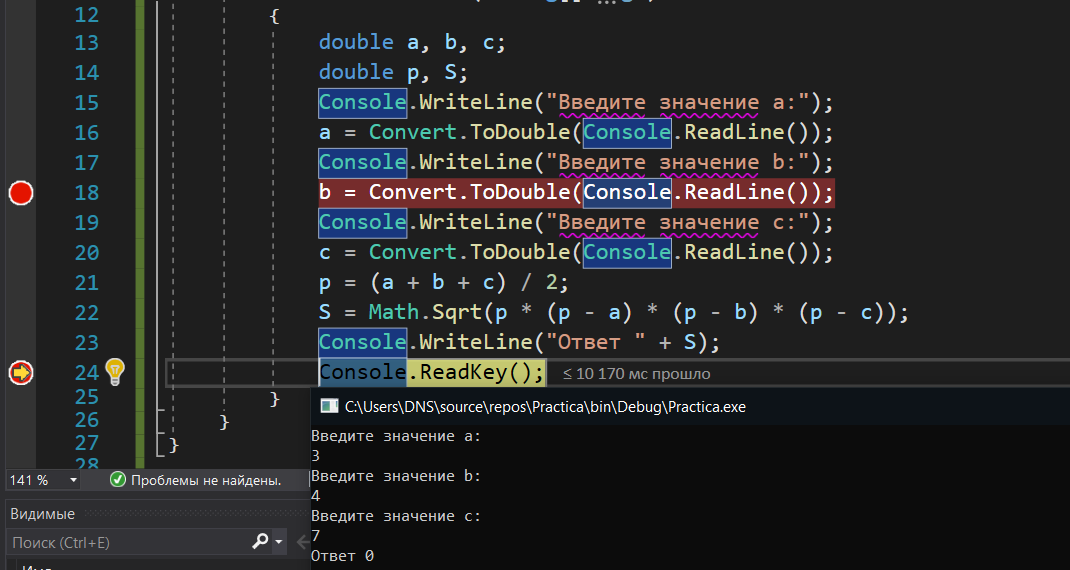


Рис.7

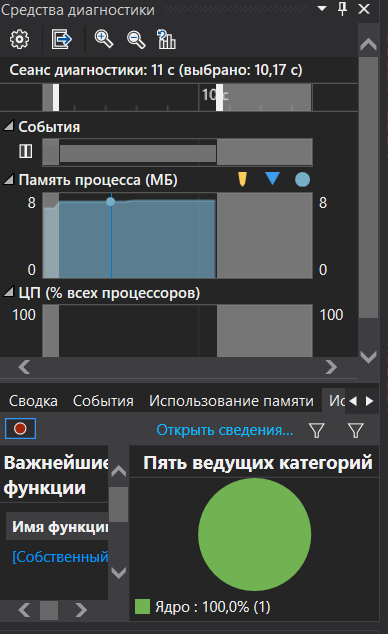


Рис.8

9. Если вы хотите выбрать для анализа более конкретную область кода, выберите область на временной шкале ЦП (это должна быть область, в которой отображаются данные профилирования).

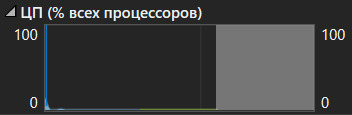


Рис.9

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

1. В списке функций дважды щелкните одну из функций вашего приложения, которая выполняет много работы.

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

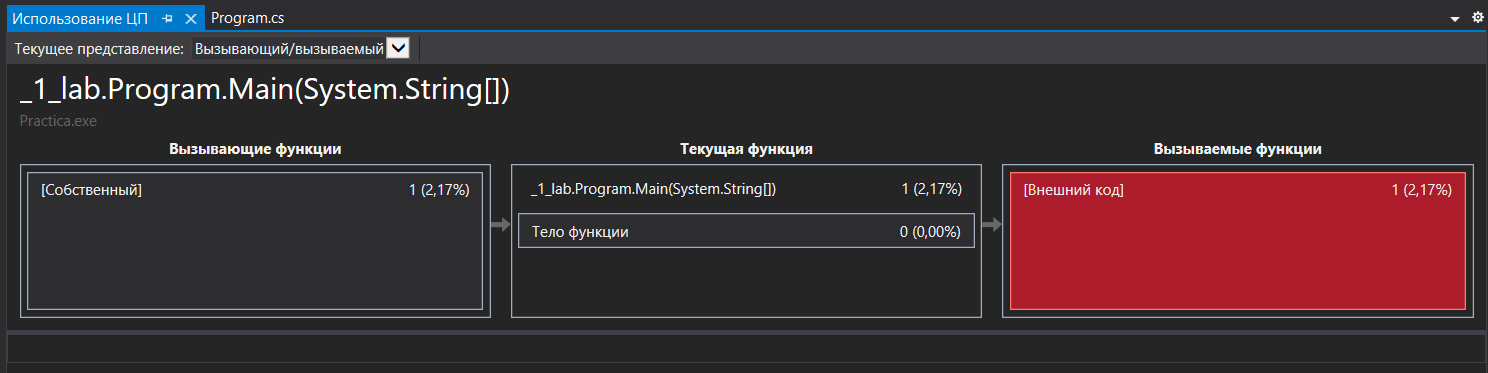


Рис.10

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

Каждая нумерованная область на рисунке соответствует определенному шагу в процедуре.

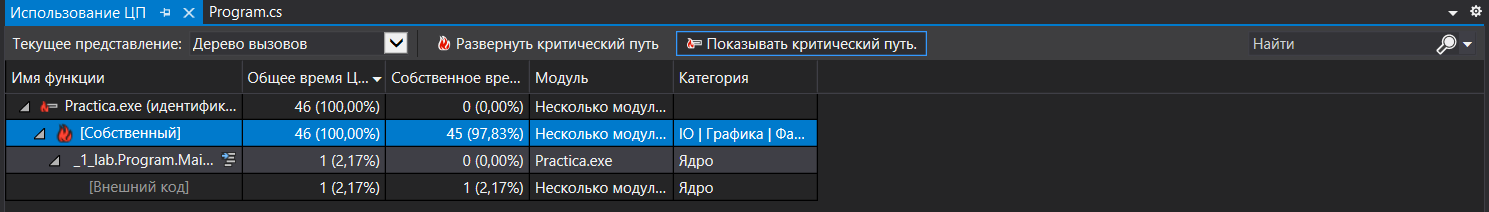


Рис.11

Внешний код — это функции в компонентах системы и платформы, которые исполняются вашим кодом. Внешний код включает функции, которые запускают и останавливают приложение, отрисовывают пользовательский интерфейс, управляют потоками и предоставляют приложению другие низкоуровневые службы. В большинстве случаев внешний код вас интересовать не будет, поэтому средство "Загрузка ЦП" собирает внешние функции пользовательского метода в один узел **[Внешний код]** .

Если вы захотите посмотреть пути к вызовам внешнего кода, выберите **Показать внешний код** в списке **Представление фильтра** и выберите **Применить**.

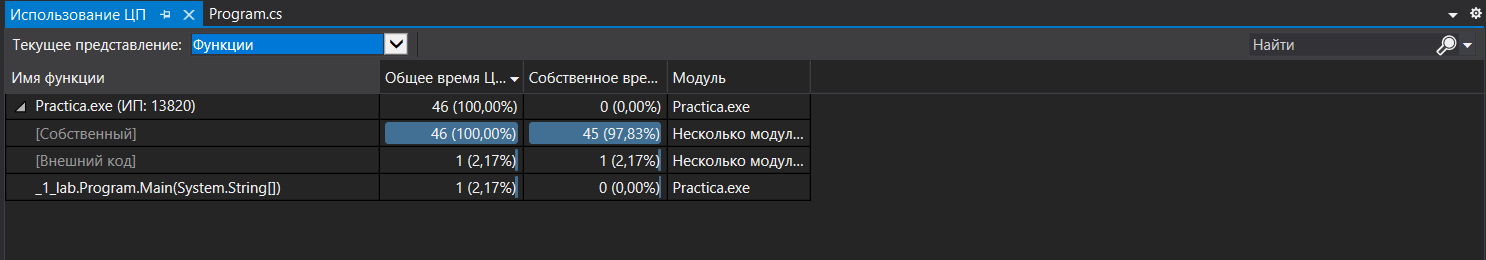


Рис.12