

Лабораторная работа №15

Тестирование, отладка приложения

1 Цель работы

1.1 Изучить возможности отладки приложений в Visual Studio.

2 Литература

2.1 Документация по отладчику — Visual Studio. – Текст : электронный // Microsoft Learn : официальный сайт. – 2024. – URL: <https://learn.microsoft.com/ru-ru/visualstudio/debugger/?view=vs-2022> (дата обращения 18.09.2024).

3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

5.1 Пошаговая отладка и просмотр значений переменных

Создать консольное приложение, содержащее цикл со счетчиком, в котором вычисляется значение факториала f от программно заданного числа n . Результат должен выводиться на экран. Указать $n=6$ и выполнить пошаговую отладку с отслеживанием значений переменных во вкладке Локальные. Реализовать отображение данных во время пошаговой отладки.

5.2 Создание и отключение брейкпоинтов (точек останова)

Создать консольное приложение, дублирующее приложение из п.5.1. Добавить брейкпоинт в конце цикла (после изменения значения переменной f) и запустить отладку с остановками на брейкпоинтах с отслеживанием значений переменных во вкладке Локальные. Сделать имеющийся брейкпоинт неактивным (Ctrl+F9).

5.3 Настройка остановки по условию

Создать консольное приложение, дублирующее приложение из п.5.1. Добавить брейкпоинт в конце цикла (после изменения значения переменной f). Изменить брейкпоинт так, чтобы остановки выполнялись только если i нечетное или $f > 30$. Запустить отладку с остановками на брейкпоинтах с отслеживанием значений переменных во вкладке Локальные.

5.4 Настройка действия у брейкпоинта

Создать консольное приложение, дублирующее приложение из п.5.1. Добавить брейкпоинт в конце цикла (после изменения значения переменной f) и запустить отладку с остановками на брейкпоинтах с отслеживанием значений переменных во вкладке Локальные.

Изменить брейкпоинт так, чтобы при его достижении в отладочную консоль (окно Вывод) выводились значения i и f в следующем виде:

$i! = f$

Образец вывода:

$0! = 1$

$1! = 1$

$2! = 2$

$3! = 6$

5.5 Отладка с заходом в функции

Создать консольное приложение, дублирующее приложение из п.5.1.

Провести рефакторинг, выделив функцию для вычисления факториала из функции `main`. Значение `n` в основной программе должно запрашиваться у пользователя, функция вычисления факториала должна возвращать число (в случае некорректных данных должен вернуть -1).

Установить брейкпоинт на вызове метода вычисления факториала из функции `main` и выполнить пошаговую отладку с заходом в функцию.

6 Порядок выполнения работы

6.1 Используя Microsoft Visual Studio, создать проект C++ и выполнить задания из п.5.

6.2 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Что такое «отладка»?

8.2 Какие этапы включает в себя отладка?

8.3 Что такое «точка останова»?

8.4 Как добавить точки останова в Visual Studio?

8.5 Как в Visual Studio запустить приложение в режиме отладки?

8.6 Какие окна в Visual Studio отображают значения переменных и для чего предназначено каждое из этих окон?

8.7 Какие комбинации «горячих» клавиш используются в Visual Studio для пошагового прохождения?