**BC/NW 2024 № 1 (41):12.3**

**СТАТИЧЕСКИЙ АНАЛИЗ ВЗАИМНЫХ БЛОКИРОВОК С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ СИМВОЛЬНОГО АНАЛИЗА В C++**

Скоробогатов Д.Г., Зайнутдинов М.М., Хиль С.Ю., Орлов Д.А

В современной разработке программных продуктов многопоточность играет важнейшую роль, позволяя выполнять задачи параллельно, повышая производительность. Это становится особенно актуально, учитывая, что современные процессоры наращивают число физических ядер в системах.

В связи с появлением многопоточных программных продуктов возникли и новые проблемы: взаимные блокировки, состояния гонки, ресурсное голодание, инверсия приоритетов и другие. Для решения данных проблем созданы статические анализаторы, которые позволяют без компиляции кода выявить ошибки.

В рамках исследования проблемы выявления взаимных блокировок были использованы следующие статические анализаторы для языка C++: CLion, CPPcheck, PVS-studio и KLEE [1]. Была проведена проверка обнаружения проблем с взаимными блокировками посредством статических анализаторов ошибок с использованием интерфейса многопоточности std::thread, появившегося в C++11, а также POSIX Threads. Рассмотрены различные ситуации взаимных блокировок, по результатам анализа готовых программных решений, содержащих данную ошибку, установлено, что ни один из перечисленных статических анализаторов не определил наличие проблем.

Перспективным вариантом направления разработки и исследований в области статического анализа многопоточных приложений является выявление ошибок, связанных с многопоточным кодом, посредством символьных статических анализаторов, которые используют символьное выполнение программ, учитывая все возможные значения переменных и пути выполнения кода, а также сценарии взаимодействия между потоками.

Таким образом, задача разработки или модификации готовых решений имеющихся инструментов для символьного статического анализа программ, ориентированных на обнаружение ошибок, связанных с многопоточностью, является актуальной.

**Литература**

1. **А.Г. Зыков, И.В. Кочетков, В.И. Поляков.** Применение системы KLEE для автоматизации тестирования программ на языках C/C++. Научно-практический журнал “Программные продукты и системы” – 14.06.2016 – С. 101 – 106.