004.021
# Методы проверки корректности сборщиков мусора
Авторы: Кузенкова Е.В., Кореньков Ю. Д., Логинов И.П., Университет ИТМО, Санкт-Петербург
Научный руководитель: Кореньков Ю. Д. Университет ИТМО, Санкт-Петербург

Многие современные языки программирования доверяют управление памятью сборщику мусора, который становится важной частью любой современной среды выполнения. Сборщик мусора – программа, которая запускается, когда при очередном выделении памяти аллокатор выявляет, что недостаточно свободного места для совершения текущей аллокации. Задача сборщика мусора – найти свободное место путем удаления недостижимых объектов и перемещении занятых участков памяти. Есть два классических алгоритма: пометить-и-собрать (mark-and-sweep) и копировать.

Существует огромное количество различных усовершенствований и комбинаций этих алгоритмов, однако не все они имеют доказательства корректности своей работы. Целью нашей работы является рассмотрение возможных методов проверки корректности сборщиков мусора.

Можно выделить два основных способа решения задачи доказательства корректности алгоритма. Формальный способ доказательства включает в себя формализацию алгоритма и его формальное доказательство на таких языках как Coq, B и др. Инженерный подход состоит в написании тестов для оценки корректности работы сборщика мусора и проверки их выполнимости.

В рамках данной работы рассматривается проверка корректности реализованного нами сборщика мусора по алгоритму пометил-и-собрал. Так как данный сборщик мусора был предназначен для экспериментальной платформы, первым делом был проведён анализ различных существующих подходов к сборке мусора и сформулирован публичный контракт сборщика мусора – набор программных интерфейсов, обеспечивающих не только возможность замены сборщиков мусора, используемых тестовым либо прикладным окружением, но и возможность реализации в рамках такого программного интерфейса сборщиков мусора с различными стратегиями сборки. Для этого были выделены основные действия приложения по работе с объектами в управляемой куче – области памяти, управляемой аллокатором, снабженным сборщиком мусора, и имеющим доступ к метаданным, описывающим внутреннюю структуру аллоцируемых блоков памяти, необходимым для любого не консервативного алгоритма сборки мусора. Основные из таких действий: аллокация объекта, создание корня объектнго графа, удаление корня, создание и удаление ссылки на достижимый объект в куче, завершение работы программы.

Для разработанных программных интерфейсов и внутреннего состояния кучи сборщика мусора были сформулированы инварианты такие как: удаляемый объект должен быть недостижим из корней, все корни должны быть помечены и др., и для каждого из таких правил введены верифицирующие проверки в составе тестового окружения. В результате был создан набор правил-ограничений корректности, при успешном выполнении которых в соответствующих точках логики сборщика мусора и на границах программных интерфейсов мы можем утверждать, что сборщик мусора работает корректно.

Разработанный сборщик мусора был успешно проверен на соответствие созданным наборам правил, ведётся работа по созданию ещё одного сборщика мусора в рамках созданных программных интерфейсов для практической проверки применимости разработанных тестов к различным методам сборки.