

УДК 004.438.NET  
ББК 32.973.26-018.2  
К55

Кокоса К.  
К55 Управление памятью в .NET для профессионалов. – М.: ДМК Пресс, 2020. – 800 с.: ил.

**ISBN 978-5-97060-800-5**

Хотя в .NET управление памятью осуществляется автоматически, понимание того, как именно это делается, сулит немало преимуществ. Вы сможете писать более качественные программы, эффективно взаимодействующие с памятью. Книга содержит 25 сценариев поиска и устранения неисправностей, призванных помочь в диагностике сложных проблем при работе с памятью. Приводится также ряд полезных рекомендаций по написанию кода, учитывающих особенности управления памятью и позволяющих избежать типичных ошибок.

Книга адресована разработчикам программного обеспечения для платформы .NET, архитекторам и специалистам по производительности.

УДК 004.438.NET  
ББК 32.973.26-018.2

Original English language edition printed on acid-free paper. Copyright © 2018 by Konrad Kokosa.  
Russian language edition copyright © 2020 by DMK Press. All rights reserved.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN 978-1-4842-4026-7 (англ.)  
ISBN 978-5-97060-800-5 (рус.)

Copyright © 2018 by Konrad Kokosa  
© Оформление, издание, перевод, ДМК Пресс, 2020

# Содержание

<b>Об авторе .....</b>	<b>15</b>
<b>О технических рецензентах .....</b>	<b>15</b>
<b>Благодарности.....</b>	<b>16</b>
<b>Предисловие .....</b>	<b>18</b>
<b>Введение .....</b>	<b>19</b>
<b>От издательства.....</b>	<b>25</b>
<b>Глава 1. Основные понятия .....</b>	<b>26</b>
Терминология, относящаяся к памяти.....	27
Статическое выделение .....	33
Регистровая машина .....	34
Стек .....	35
Стековая машина .....	40
Указатель .....	43
Куча .....	45
Ручное управление памятью .....	47
Автоматическое управление памятью.....	52
Распределитель, модификатор и сборщик.....	54
Подсчет ссылок .....	58
Отслеживающий сборщик .....	63
Этап пометки.....	63
Этап сборки .....	67
Немного истории.....	71
Резюме.....	73
Правило 1: учиться, учиться и учиться.....	74
<b>Глава 2. Низкоуровневое управление памятью .....</b>	<b>75</b>
Оборудование .....	76
Память.....	81
Центральный процессор.....	84
Операционная система .....	99
Виртуальная память.....	100
Большие страницы.....	104
Фрагментация виртуальной памяти .....	105
Общая структура памяти.....	105
Управление памятью в Windows.....	107
Организация памяти в Windows .....	112

Управление памятью в Linux .....	114
Организация памяти в Linux.....	116
Зависимость от операционной системы .....	117
NUMA и группы процессоров.....	118
Резюме.....	120
Правило 2: избегайте произвольного доступа, отдавайте предпочтение последовательному .....	120
Правило 3: улучшайте пространственную и временную локальность данных.....	121
Правило 4: пользуйтесь продвинутыми средствами.....	121

## Глава 3. Измерения памяти..... 123

Измеряйте как можно раньше.....	124
Накладные расходы и вмешательство.....	125
Выборка и трассировка.....	126
Дерево вызовов .....	126
Графы объектов .....	127
Статистика .....	129
Задержка и пропускная способность.....	132
Дампы памяти, трассировка, динамическая отладка.....	133
Среда Windows .....	134
Краткий обзор .....	134
VMMap.....	135
Счетчики производительности .....	136
Трассировка событий для Windows .....	142
Windows Performance Toolkit .....	152
PerfView.....	162
ProcDump и DebugDiag.....	171
WinDbg.....	171
Дизассемблеры и декомпиляторы .....	174
BenchmarkDotNet .....	174
Коммерческие инструменты .....	176
Среда Linux.....	186
Краткий обзор .....	186
Perfcollect .....	187
Trace Compass .....	189
Дампы памяти.....	198
Резюме.....	199
Правило 5: измеряйте GC как можно раньше.....	201

## Глава 4. Фундаментальные основы .NET..... 202

Версии .NET.....	202
Детали внутреннего устройства .NET.....	205
Разбираем пример программы.....	208
Сборки и домены приложений.....	213
Забираемые сборки.....	215
Области памяти процесса .....	216
Сценарий 4.1. Сколько места в памяти занимает моя программа? .....	220
Сценарий 4.2. Моя программа потребляет все больше и больше памяти .....	222
Сценарий 4.3. Моя программа потребляет все больше и больше памяти .....	225
Сценарий 4.4. Моя программа потребляет все больше и больше памяти .....	227
Система типов.....	230

Категории типов.....	231
Хранение типов.....	232
Типы значений.....	233
Ссылочные типы .....	241
Строки .....	246
Интернирование строк.....	252
Сценарий 4.5. Моя программа потребляет слишком много памяти.....	257
Упаковка и распаковка .....	259
Передача по ссылке .....	264
Передача по ссылке экземпляра типа значений .....	264
Передача по ссылке экземпляра ссылочного типа .....	265
Локальность типов данных .....	266
Статические данные.....	269
Статические поля .....	269
Внутреннее устройство статических данных.....	270
Резюме.....	273
Структуры .....	274
Классы .....	274

## **Глава 5. Разделение памяти на части .....**

Стратегии разделения памяти.....	278
Разделение по размеру .....	279
Куча малых объектов .....	280
Куча больших объектов.....	281
Разделение по времени жизни.....	284
Сценарий 5.1. Как чувствует себя моя программа? Динамика размеров поколений ...	290
Запомненные наборы (Remembered sets).....	292
Таблицы карт (Card tables).....	298
Связки карт.....	303
Физическое разделение .....	306
Сценарий 5.2. Утечка памяти в porCommerce?.....	311
Сценарий 5.3. Растраниживание кучи больших объектов? .....	319
Анатомия сегментов и кучи .....	321
Повторное использование сегментов.....	324
Резюме.....	326
Правило 11: следите за размерами поколений.....	326
Правило 12: избегайте лишних ссылок в куче .....	327
Правило 13: наблюдайте за использованием сегментов .....	328

## **Глава 6. Выделение памяти .....**

Введение в распределение памяти .....	329
Выделение памяти сдвигом указателя.....	330
Выделение памяти из списка свободных блоков .....	337
Создание нового объекта .....	341
Выделение памяти в куче малых объектов .....	343
Выделение памяти в куче больших объектов .....	347
Балансировка кучи .....	351
Исключение OutOfMemoryException.....	353
Сценарий 6.1. Нехватка памяти .....	355
Выделение памяти в стеке .....	356
Избегание выделения памяти .....	358

Явное выделение памяти для ссылочных типов .....	360
Скрытое выделение памяти .....	381
Скрытое выделение памяти в библиотеках .....	389
Сценарий 6.2. Исследование выделения памяти .....	393
Сценарий 6.3. Функции Azure .....	396
Резюме .....	397
Правило 14: избегайте выделения памяти в куче на критических с точки зрения производительности участках программы .....	397
Правило 15: избегайте дорогостоящего выделения памяти в LOH .....	398
Правило 16: по возможности выделяйте память в стеке .....	398

## **Глава 7. Сборка мусора – введение** .....

Общее описание .....	400
Пример процесса сборки мусора .....	402
Шаги процесса сборки мусора .....	408
Сценарий 7.1. Анализ использования GC .....	408
Профилирование GC .....	412
Данные для настройки производительности сборки мусора .....	414
Статические данные .....	414
Динамические данные .....	417
Сценарий 7.2. Демонстрация бюджета выделения .....	419
Инициаторы сборки мусора .....	428
Запуск по причине выделения памяти .....	429
Явный запуск .....	430
Сценарий 7.3. Анализ явных вызовов GC .....	433
Запуск по причине нехватки памяти у системы .....	439
Запуск по различным внутренним причинам .....	439
Приостановка движка выполнения .....	440
Сценарий 7.4. Анализ времени приостановки GC .....	442
Выбор поколения для сборки .....	444
Сценарий 7.5. Анализ выбираемых поколений .....	447
Резюме .....	448

## **Глава 8. Сборка мусора – этап пометки** .....

Обход и пометка объектов .....	449
Корни – локальные переменные .....	450
Хранилище локальных переменных .....	451
Корни на стеке .....	452
Лексическая область видимости .....	452
Живые стековые корни и лексическая область видимости .....	453
Живые стековые корни с ранней сборкой корней .....	455
Информация для GC (GC Info) .....	461
Закрепленные локальные переменные .....	465
Просмотр стековых корней .....	468
Корни финализации .....	468
Внутренние корни GC .....	469
Корни – описатели GC .....	470
Анализ утечек памяти .....	476
Сценарий 8.1. Утечка памяти в porCommerce? .....	478
Сценарий 8.2. Нахождение самых популярных корней .....	482
Резюме .....	484

<b>Глава 9. Сборка мусора – этап планирования</b>	485
Куча малых объектов	486
Заполненные и пустые блоки	486
Сценарий 9.1. Дамп памяти с поврежденными структурами	491
Таблица кирпичей	492
Закрепление	494
Сценарий 9.2. Исследование закрепления	499
Границы поколений	504
Оставление	504
Куча больших объектов	509
Заполненные и пустые блоки	509
Принятие решения об уплотнении	511
Резюме	512
<b>Глава 10. Сборка мусора – очистка и уплотнение</b>	513
Этап очистки	513
Куча малых объектов	513
Куча больших объектов	514
Этап уплотнения	515
Куча малых объектов	515
Куча больших объектов	519
Сценарий 10.1. Фрагментация кучи больших объектов	520
Резюме	528
Правило 17: следите за приостановкой среды выполнения	529
Правило 18: избегайте кризиса среднего возраста	529
Правило 19: избегайте фрагментации старого поколения и LOH	530
Правило 20: избегайте явной сборки мусора	531
Правило 21: избегайте утечек памяти	531
Правило 22: избегайте закрепления	532
<b>Глава 11. Варианты сборки мусора</b>	533
Обзор режимов	533
Режим рабочей станции и серверный режим	533
Неконкурентный и конкурентный режим	535
Конфигурирование режимов	536
.NET Framework	537
.NET Core	537
Приостановка и накладные расходы GC	538
Описание режимов	540
Неконкурентный режим рабочей станции	541
Конкурентный режим рабочей станции (до версии 4.0)	542
Фоновый режим рабочей станции	544
Неконкурентный серверный режим	552
Фоновый серверный режим	554
Режимы задержки	556
Пакетный режим	556
Интерактивный режим	557
Режим низкой задержки	557
Режим длительной низкой задержки	558
Регион без сборки мусора (No GC Region)	559
Цели оптимизации задержки	562

Выбор варианта GC.....	562
Сценарий 8.1. Проверка параметров GC .....	563
Сценарий 8.2. Измерение и тестирование производительности различных режимов GC .....	566
Резюме.....	573
Правило 23: выбирайте режим GC обдуманно .....	573
Правило 24: помните о режимах задержки.....	574
<b>Глава 12. Время жизни объекта .....</b>	<b>575</b>
Жизненные циклы объекта и ресурса.....	575
Финализация.....	577
Введение .....	577
Проблема ранней сборки корней.....	582
Критические финализаторы .....	585
Внутреннее устройство финализации.....	586
Сценарий 12.1. Утечка памяти из-за финализации.....	593
Воскрешение .....	599
Уничтожаемые объекты .....	603
Безопасные описатели .....	609
Слабые ссылки .....	614
Кеширование .....	618
Паттерн слабых событий .....	620
Сценарий 9.2. Утечка памяти из-за событий .....	626
Резюме.....	629
Правило 25: избегайте финализаторов .....	629
Правило 26: отдавайте предпочтение явной очистке .....	630
<b>Глава 13. Разное .....</b>	<b>632</b>
Зависимые описатели .....	632
Локальная память потока .....	638
Статические поля потока.....	638
Слоты данных потока .....	641
Внутреннее устройство локальной памяти потока .....	642
Сценарии использования .....	649
Управляемые указатели .....	650
Ссылочные локальные переменные .....	651
Возвращаемые ссылочные значения.....	652
Постоянные ссылочные переменные и in-параметры.....	654
Внутреннее устройство ссылочных типов .....	658
Управляемые указатели в C# – ссылочные переменные.....	669
И снова о структурах .....	675
Постоянные структуры .....	676
Ссылочные структуры (byref-подобные типы).....	677
Буферы фиксированного размера .....	679
Размещение объектов и структур в памяти .....	683
Ограничение unmanaged.....	694
Непреобразуемые типы .....	698
Резюме.....	700
<b>Глава 14. Продвинутое приемы .....</b>	<b>701</b>
Span<T> и Memory<T> .....	701
Span<T> .....	702

## 12 ❖ Содержание

Memory<T> .....	716
IMemoryOwner<T> .....	719
Внутреннее устройство Memory<T>.....	723
Рекомендации по работе с Span<T> и Memory<T> .....	725
Класс Unsafe .....	726
Внутреннее устройство Unsafe .....	730
Проектирование, ориентированное на данные .....	731
Тактическое проектирование .....	732
Стратегическое проектирование .....	736
Еще немного о будущем.....	745
Ссылочные типы, допускающие null .....	746
Конвейеры .....	751
Резюме.....	757

## **Глава 15. Интерфейсы прикладного программирования (API)..... 759**

GC API .....	759
Сведения и статистические данные о сборке мусора.....	760
Уведомления GC .....	768
Контроль потребления неуправляемой памяти .....	770
Явная сборка мусора .....	770
Области без GC .....	770
Управление финализацией .....	770
Потребление памяти.....	771
Внутренние вызовы в классе GC .....	772
Размещение CLR .....	773
ClrMD .....	782
Библиотека TraceEvent .....	787
Пользовательский сборщик мусора .....	790
Резюме.....	793

## **Предметный указатель..... 795**