Contents

Foreward, by John L. Hennessy **vii**

Preface **xiii**

Chapter 1 **Introduction** **1**

1.1 Performance with OpenMP 2

1.2 A First Glimpse of OpenMP 6

1.3 The OpenMP Parallel Computer 8

1.4 Why OpenMP? 9

1.5 History of OpenMP 13

1.6 Navigating the Rest of the Book 14

Chapter 2 **Getting Started with OpenMP** **15**

2.1 Introduction 15

2.2 OpenMP from 10,000 Meters 16

2.2.1 OpenMP Compiler Directives or Pragmas 17

2.2.2 Parallel Control Structures 20

2.2.3 Communication and Data Environment 20

2.2.4 Synchronization 22

2.3 Parallelizing a Simple Loop 23

2.3.1 Runtime Execution Model of an OpenMP Program 24

2.3.2 Communication and Data Scoping 2.5

2.3.3 Synchronization in the Simple Loop Example 27
2.3.4 Final Words on the Simple Loop Example 28
2.4 A More Complicated Loop 29
2.5 Explicit Synchronization 32
2.6 The reduction Clause 35
2.7 Expressing Parallelism with Parallel Regions 36
2.8 Concluding Remarks 39
2.9 Exercises 40

Chapter 3

Exploiting Loop-Level Parallelism 41

3.1 Introduction 41
3.2 Form and Usage of the *parallel* do Directive 42
3.2.1 Clauses 43
3.2.2 Restrictions on Parallel Loops 44
3.3 Meaning of the *parallel* do Directive 46
3.3.1 Loop Nests and Parallelism 46
3.4 Controlling Data Sharing 47
3.4.1 General Properties of Data Scope Clauses 49
3.4.2 The *shared* Clause 50
3.4.3 The *private* Clause 51
3.4.4 Default Variable Scopes 53
3.4.5 Changing Default Scoping Rules 56
3.4.6 Parallelizing Reduction Operations 59
3.4.7 Private Variable Initialization and Finalization 63
3.5 Removing Data Dependences 65
3.5.1 Why Data Dependences Are a Problem 66
3.5.2 The First Step: Detection 67
3.5.3 The Second Step: Classification 71
3.5.4 The Third Step: Removal 73
3.5.5 Summary 81
3.6 Enhancing Performance 82
3.6.1 Ensuring Sufficient Work 82
3.6.2 Scheduling Loops to Balance the Load 85
3.6.3 Static and Dynamic Scheduling 86
3.6.4 Scheduling Options 86
3.6.5 Comparison of Runtime Scheduling Behavior 88
3.7 Concluding Remarks 90
3.8 Exercises 90

Chapter 4 **Beyond Loop-Level Parallelism: Parallel Regions** **93**

4.1 Introduction 93

4.2 Form and Usage of the *parallel* Directive 94

4.2.1 Clauses on the *parallel* Directive 95

4.2.2 Restrictions on the *parallel* Directive 96

4.3 Meaning of the *parallel* Directive 97

4.3.1 Parallel Regions and SPMD-Style Parallelism 100

4.4 *threadprivate* Variables and the *copyin* Clause 100

4.4.1 The *threadprivate* Directive 103

4.4.2 The *copyin* Clause 106

4.5 Work-Sharing in Parallel Regions 108

4.5.1 A Parallel Task Queue 108

4.5.2 Dividing Work Based on Thread Number 109

4.5.3 Work-Sharing Constructs in OpenMP 111

4.6 Restrictions on Work-Sharing Constructs 119

4.6.1 Block Structure 119

4.6.2 Entry and Exit 120

4.6.3 Nesting of Work-Sharing Constructs 122

4.7 Orphaning of Work-Sharing Constructs 123

4.7.1 Data Scoping of Orphaned Constructs 125

4.7.2 Writing Code with Orphaned Work-Sharing Constructs 126

4.8 Nested Parallel Regions 126

4.8.1 Directive Nesting and Binding 129

4.9 Controlling Parallelism in an OpenMP Program 130

4.9.1 Dynamically Disabling the *parallel* Directives 130

4.9.2 Controlling the Number of Threads 131

4.9.3 Dynamic Threads 133

4.9.4 Runtime Library Calls and Environment Variables 135

4.18 Concluding Remarks 137

4.11 Exercises 138

Chapter 5 **Synchronization** **141**

5.1 Introduction 141

5.2 Data Conflicts and the Need for Synchronization 142

5.2.1 Getting Rid of Data Races 143

5.2.2 Examples of Acceptable Data Races 144
5.2.3 Synchronization Mechanisms in OpenMP 146
5.3 Mutual Exclusion Synchronization 147
5.3.1 The Critical Section Directive 147
5.3.2 The *atomic* Directive 152
5.3.3 Runtime Library Lock Routines 155
5.4 Event Synchronization 157
5.4.1 Barriers 157
5.4.2 Ordered Sections 159
5.4.3 The *muster* Directive 161
5.5 Custom Synchronization: Rolling Your Own 162
5.5.1 The *flush* Directive 163
5.6 Some Practical Considerations 165
5.7 Concluding Remarks 168
5.8 Exercises 168

Chapter 6 **Performance** **171**

6.1 Introduction, 171
6.2 Key Factors That Impact Performance 173
6.2.1 Coverage and Granularity 173
6.2.2 Load Balance 175
6.2.3 Locality *179*
6.2.4 Synchronization 192
6.3 Performance-Tuning Methodology 198
6.4 Dynamic Threads 201
6.5 Bus-Based and NUMA Machines 204
6.6 Concluding Remarks 207
6.7 Exercises 207

Appendix A A Quick Reference to OpenMP **211**
References **217**
Index **221**