Preface

Recent advances in Internet and Web are changing the way we conduct business, manage our life, and interact among ourselves as a society. They have made the world a global village for information exchange and service delivery. However, developing software systems and applications for these environments continues to be a complex and challenging task. In addition, the cost of software maintenance is increasing at a rapid pace surpassing the cost of its development and hardware used for running it. Several paradigms and methodologies have been developed to manage this software crisis. Object-Oriented Programming (OOP) has emerged as the most popular silver bullet for managing complexity associated with the development and maintenance of software systems and applications.

Several object-oriented programming languages have been invented since 1960. The two most well-known ones are: C++ and Java. The emergence of Web as media for information exchange and service delivery in early 1990s has created the need for a programming language supporting networked environments involving a wide variety of computers and devices. To meet these requirements, Sun Microsystems developed the Java programming language, which has rapidly emerged as a dominant OOP language for implementing Web and Internet service applications. As a platform independent language, Java provides capabilities such as network, graphic, and concurrent programming as its core elements.

Coverage and Resources

The "Object Oriented Programming with Java: Essentials and Applications" book introduces the software crisis the industry is facing due to the challenges associated with the development and maintenance of large-scale software systems and applications. Then it presents OOP as a solution with Java as a programming language. The book covers fundamentals of OOP and Java programming at both basic and advanced levels. It offers a balanced treatment of OOP theory and practice for developing desktop, enterprise, and web applications. These features make it a unique textbook for both undergraduate and postgraduate students. The advanced topics covered include Socket programming, multithreading, GUI (Graphical User Interface) programming, RMI (Remote Method Invocation), JDBC (Java Database Connectivity), Java Servlet, JavaServer Pages and Java Beans. Such coverage ensures that the book also serves as a reference for software engineers and practitioners working in IT and other industries.

Every chapter comes with an extensive set of exercises: objective questions, review questions, and programming problems. We encourage students to try these out by themselves to test and enhance their understanding of the subject. However, we have included answers to objective questions only (see Appendix D) - just for verification purpose!

To encourage students to put all concepts learned in this book into practice, we have proposed two projects: Automation of a Publishing House and a Bank - complete details on these projects are included in Appendix A and B.

To enrich teaching and learning experience using this book, we have created a Web Resource Center providing pointers/links to online resources, educational materials such as presentation slides, white papers detailing recent advances, and innovative web applications. For details, please visit the book's website:

```
http://www.buyya.com/java/
```

Acknowledgments

First and foremost, we are grateful to all of our colleagues for contributing their time, effort, and understanding during the preparation of the book. They include: Selina Dennis, Shanika Karunasekera, Christian Vecchiola, Charity Laplap, Suraj Pandey, and Rodrigo Calheiros. We offer our sincere gratitude to our employers for their support and cooperation.

We thank members of the GRIDS Lab for proofreading one or more chapters. They include Rajiv Ranjan, James Broberg, Chee Shin Yeo, Alexandre di Costanzo, Srikumar Venugopal, Marco Netto, Mukaddim Pathan, Ming Zhu, Mudiyanselage Wickremasinghe, Mustafizur Rahman, Saurabh Garg, William Voorsluys, Mohsen Amini, Amir Vahid, Arun Anandasivam and Anton Beloglazov.

We would like to thank all of our colleagues at Melbourne University who taught Software Design subject as their teaching materials have influenced on the content of this book. Some of the contents of this book have evolved over a period of time from our own teaching of subjects such as Distributed Systems and Grid Computing. We would like to thank Rao Kotagiri for his mentorship and support in mounting courses in these areas.

We thank our family members, especially Smrithi Buyya, Soumya Buyya, Radha Buyya, Siyin Sun for their love and understanding during the preparation of the book.

We sincerely thank external reviewers commissioned by the publisher for their critical comments and suggestions on enhancing the presentation and organisation of many chapters at a finer level. This has greatly helped us in improving the quality of the book.

Finally, we would like to thank the staff at McGraw Hill Education (I) Press for their enthusiastic support and guidance during the preparation of the book. In particular, Vibha Mahajan inspired us to take up this project and set the publication process in motion, Nilanjan Chakravarty managed the manuscript review process, and Surbhi Suman guided us in updating the book to enrich the content and ensured that it covers topics prescribed in the syllabus of many educational institutions. They were wonderful to work with!

Rajkumar Buyya
The University of Melbourne and Manjrasoft Pvt Ltd, Australia

Thamarai Selvi Somasundaram
Anna University Chennai, India

Xingchen Chu
The University of Melbourne, Australia