Chapter 8

A New Industry–Centred Module on Structured Parallel Programming

Horacio González-Vélez
Robert Gordon University, UK

ABSTRACT

This work presents the case for the introduction of a new module on parallel programming for the core degree programmes in the School of Computing at the Robert Gordon University, and elsewhere. Having been conceived and designed with the industry-leading tools for structured parallel programming in mind, this module introduces students to parallel architectures, structured parallelism, and parallel programming. The main innovation of our approach is its emphasis on the structured parallelism environments recently released by Google, Microsoft, and Intel.

INTRODUCTION

Knowledge transfer between universities and industries has long been nurtured by different institutional and governmental bodies across the world. While there is an important transfer through journal articles, conferences, projects and dedicated events, the bulk of the conveyance in computer and information science is achieved through university graduates who get degrees and, subsequently, are employed in industry and some of whom return to academia.

Computer science graduates must therefore possess not only solid theoretical foundations, but also transferable job skills and experience to be employable and attractive to the information and communications technology (ICT) industry. Human resource requirements in industry must be swiftly incorporated into the curricula, be-

cause otherwise graduates risk relegation from the mainstream.

The advent of multi-core processors has strongly impacted industry, as most personal computers now feature multi-core chips and many mobile devices feature low-power multiprocessing units. Furthermore, chip multiprocessors, multi-node clusters, grids, and clouds have sharply increased the number of concurrent processors available for a single application. Accordingly, industry increasingly requires software professionals with parallel programming skills.

Parallel programming aims to capitalise on concurrency, the execution of different sections of a given program at the same time, in order to improve the overall performance of the program, and, eventually, that of the whole system. Despite major breakthroughs, parallel programming is still a highly demanding activity widely acknowledged to be more difficult than its sequential counterpart, and one for which the use of efficient programming models and structures has long been sought. These programming models must necessarily be performance-oriented, and are expected to be defined in a scalable structured fashion to provide guidance on the execution of their jobs and assist in the deployment of heterogeneous resources and policies.

This work presents a proposal for the development of an introductory module on parallel programming, where the main emphasis is on using structured parallelism with an industry-centred approach. Such a module is intended to be incorporated in the undergraduate and post-graduate programmes in the School of Computing at the Robert Gordon University, and in similar programmes internationally.

BACKGROUND

As part of a panel of experts at a recent computer science conference, Ivanov et al. (2008) high-lighted the key factors in the case for teaching parallel computing in universities:

- **Essential**: Parallel computing provides a virtual laboratory for many disciplines such as systems biology, physics, astronomy, meteorology, sociology and anthropology, enabling them to focus on complex problems that would be otherwise unsolvable.
- **Available and Affordable**: Parallel computing is here to stay and expand: every single device is expected to feature parallelism, and some supercomputing facilities have machines with hundreds of thousands of cores. This trend is not expected to change but to permeate still more parts of our lives.
- **Difficult**: Computing has been traditionally taught primarily within a sequential frame of mind, but parallel solutions require a different way of approaching and dissecting a problem. They require a holistic analysis and understanding of the system architecture, the programming paradigm, and the problem constraints. Parallel computing requires calculations to be synchronised, staged, and/or communicated over a number of different phases. Message-passing, threads, load-balancing and semaphores are matters restricted to the expert software developers.

Thus, while we know that parallel computing is a *sine qua non* for many disciplines and is here to stay, the *status quo* tends to render it an experts-only field. But teaching parallel computing ought to be no longer a question of *if*, but of *how soon*. Goth (2009) mentions that computer scientists on university faculties say academia is debating how and when to introduce parallel programming throughout the curriculum, instead of just offering an upper-level course as is now common. It is of interest that the main sources

Related Content

Progression of UK Women Engineers: Aids and Hurdles

Haifa Takruri-Rizk, Natalie Sappletonand Sunrita Dhar-Bhattacharjee (2010). *Women in Engineering, Science and Technology: Education and Career Challenges (pp. 280-300).*

www.irma-international.org/chapter/progression-women-engineers/43212

Problem-Based Learning and Authentic Assessment as an Implementation of Outcomes-Based Education in the Computer Engineering Program of Ateneo de Davao University

Eufemia Fallerand Edicio Faller (2017). *International Journal of Quality Assurance in Engineering and Technology Education (pp. 22-36).*

www.irma-international.org/article/problem-based-learning-and-authentic-assessment-as-an-implementation-of-outcomes-based-education-in-the-computer-engineering-program-of-ateneo-de-davao-university/221382

Project-Based Learning in Chemical Engineering: Curriculum and Assessment, Culture and Learning Spaces

John Robinsonand Daniel Beneroso (2022). *Training Engineering Students for Modern Technological Advancement (pp. 1-19).*

www.irma-international.org/chapter/project-based-learning-in-chemical-engineering/293557

A Brief History of Networked Classrooms to 2013: Effects, Cases, Pedagogy, and Implications with New Developments

Louis Abrahamsonand Corey Brady (2014). *International Journal of Quality Assurance in Engineering and Technology Education (pp. 1-51).*

www.irma-international.org/article/a-brief-history-of-networked-classrooms-to-2013/134452

Curriculum Issues in Industry Oriented Software Engineering Education

 (2011). *Software Industry-Oriented Education Practices and Curriculum Development: Experiences and Lessons (pp. 153-165).*

www.irma-international.org/chapter/curriculum-issues-industry-oriented-software/54979